

(12) NACH DEM VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES
PATENTWESENS (PCT) VERÖFFENTLICHTE INTERNATIONALE ANMELDUNG

(19) Weltorganisation für geistiges Eigentum
Internationales Büro



(43) Internationales Veröffentlichungsdatum
12. Mai 2005 (12.05.2005)

PCT

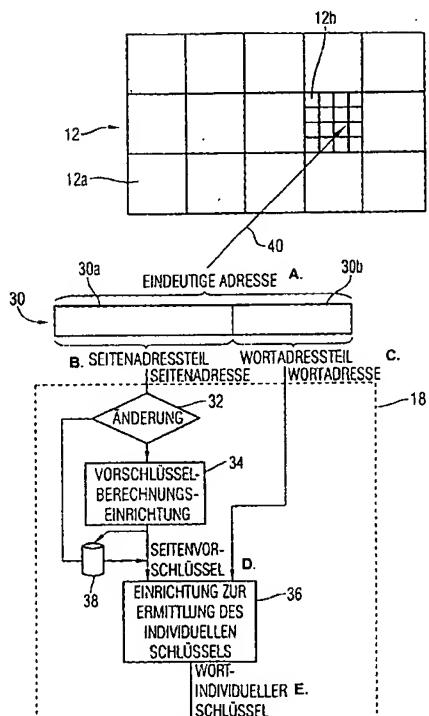
(10) Internationale Veröffentlichungsnummer
WO 2005/043396 A2

- (51) Internationale Patentklassifikation⁷: **G06F 12/14**
- (21) Internationales Aktenzeichen: PCT/EP2004/009054
- (22) Internationales Anmeldedatum:
12. August 2004 (12.08.2004)
- (25) Einreichungssprache: Deutsch
- (26) Veröffentlichungssprache: Deutsch
- (30) Angaben zur Priorität:
103 45 454.3 30. September 2003 (30.09.2003) DE
- (71) Anmelder (*für alle Bestimmungsstaaten mit Ausnahme von US*): INFINEON TECHNOLOGIES AG [DE/DE]; St.-Martin-Str. 53, München, 81669 (DE).
- (72) Erfinder; und
- (75) Erfinder/Anmelder (*nur für US*): ELBE, Astrid [DE/DE]; Am Stadtpark 40B, 81243 München (DE). GAMMEL, Berndt [DE/DE]; Dr. Brenner Str. 16, 85570 Markt Schwaben (DE). GÖTTFERT, Rainer [DE/DE]; Lachenmeyrstrasse 18A, 81827 München (DE). SONNEKALB, Steffen [DE/DE]; Kreillerstr. 51, 81673 München (DE).
- (74) Anwälte: SCHOPPE, Fritz usw.; Postfach 246, 82043 Pullach bei München (DE).
- (81) Bestimmungsstaaten (*soweit nicht anders angegeben, für jede verfügbare nationale Schutzrechtsart*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DK, DM, DZ, EC, EE, EG, ES, FI

[Fortsetzung auf der nächsten Seite]

(54) Titel: WORD-INDIVIDUAL KEY GENERATION

(54) Bezeichnung: WORTINDIVIDUELLE SCHLÜSSELERZEUGUNG



A... DISTINCT ADDRESS

B... PAGE ADDRESS PORTION PAGE ADDRESS

C... WORD ADDRESS PORTION WORD ADDRESS

32... MODIFICATION

34... PRE-KEY CALCULATION DEVICE

D... PAGE PRE-KEY

36... DEVICE FOR DETERMINING THE INDIVIDUAL KEY

E... WORD-INDIVIDUAL KEY

(57) Abstract: The invention is based on the finding that the grouping of individually addressable units of a memory (12) into groups or pages (12a) already present in many systems can be used for substantially reducing the complexity of the address-dependent key generation while only insubstantially decreasing security, if a page pre-key is calculated (36) on the basis of a page address (30a) and the individual key is determined on the basis of the page pre-key and the word address (30b). As a result, the address-dependent key generation can be split up into a cryptographically demanding and relatively time-consuming process to be carried out rarely, namely page pre-key generation (34), and a fast step (36) of virtually low technical complexity which has to be carried out for every word or every individually addressable unit (12b), namely determination of the individual key on the basis of the page pre-key and the word address (30b).

WO 2005/043396 A2

[Fortsetzung auf der nächsten Seite]



GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SI, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CR, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Veröffentlicht:

- ohne internationalen Recherchenbericht und erneut zu veröffentlichen nach Erhalt des Berichts

(84) Bestimmungsstaaten (soweit nicht anders angegeben, für jede verfügbare regionale Schutzrechtsart): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), eurasisches (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), europäisches (AT, BE, BG, CH, CY, CZ, DE, DK,

Zur Erklärung der Zweibuchstaben-Codes und der anderen Abkürzungen wird auf die Erklärungen ("Guidance Notes on Codes and Abbreviations") am Anfang jeder regulären Ausgabe der PCT-Gazette verwiesen.

(57) **Zusammenfassung:** Vorliegende Erkenntnis ist, dass die in vielen Systemen bereits vorhandene Gruppierung der individuell adressierbaren Einheiten eines Speichers (12) zu Gruppen bzw. Seiten (12a) verwendet werden kann, um die Komplexität der adressabhangigen Schlüsselgenerierung bei nur geringfügiger Verringerung an der Sicherheit deutlich zu verkleinern, wenn zunächst ein Seitenvorschlüssel auf der Basis einer Seitenadresse (30a) berechnet (36) wird und dann auf der Basis des Seiten-vorschlüssels und der Wortadresse (30b) erst der individuelle Schlüssel ermittelt wird. Hierdurch kann die adressabhängige Schlüsselgenerierung in einen kryptographisch anspruchsvollen relativ aufwendigen Prozess, der aber nur selten durchgeführt werden muss, nämlich die Seitenvorschlüsselberechnung (34), und in einen schnellen, nahezu aufwandslosen Schritt (36), der für jedes Wort bzw. jede individuell adressierbare Einheit (12b) ausgeführt werden muss, nämlich die Ermittlung des individuellen Schlüssels auf der Basis des Seitenvorschlüssels und der Wortadresse (30b), aufgespalten werden.

Beschreibung**Wortindividuelle Schlüsselerzeugung**

5 Die vorliegende Erfindung bezieht sich auf den Schutz von Speicherinhalten durch Verschlüsselung im allgemeinen und insbesondere auf die Erzeugung einheitenindividueller Schlüssel zum Zugreifen auf die adressierbaren Einheiten eines Speichers.

10 Zum Schutz gegen unberechtigtes Ausspähen von gespeicherten Informationen werden in verschiedenen Anwendungen die Speicherinhalte des Speichers verschlüsselt. Im Bereich des bargeldlosen Zahlungsverkehrs beispielsweise werden auf

15 Chipkarten gespeicherte Geldbeträge verschlüsselt gespeichert, um sie vor unberechtigtem Ausspähen zu schützen, oder vor Manipulationen, wie z.B. unberechtigten Betragsveränderungen.

20 An die in einem Speicher gespeicherten verschlüsselten Informationen, d.h. den Klartext, gelangt ein Unberechtigter beispielsweise durch statistische Analyse des auf dem Speicher gespeicherten Chiffrats. Diese statistische Analyse umfasst beispielsweise eine Analyse der Auftretenswahrscheinlichkeit bestimmter Chiffrat-Datenblöcke oder dergleichen. Um diese statistischen Analysen zu erschweren, ist es wünschenswert, dass gleiche Klartexte, die sich in verschlüsselter Form an unterschiedlichen Speicherpositionen des Speichers befinden, dort nicht in Form identischer Chiffretexte vorliegen.

30 Eine Möglichkeit, die Verschlüsselung von Klartexten an unterschiedlichen Speicherpositionen in den unterschiedlichen Chiffretexten sicherzustellen, besteht darin, zur Verschlüsselung das sogenannte Cipher-Block-Chaining-Verfahren zu verwenden, d.h. das Betreiben einer Blockchiffre im CBC-Modus, wie er beispielsweise im Handbook of Applied Cryp-

tography, CRC Press, NY, 1997, S. 230, beschrieben ist. Im
CBC-Modus wird zur Verschlüsselung eines Klartext-Datenblocks
immer das Chiffraut des vorhergehenden Klartext-Datenblocks
herangezogen, wie z.B. des Klartext-Datenblocks mit um 1
5 niedrigerer oder um 1 höherer Adresse in dem Speicher. Der
CBC-Modus hat den Nachteil, dass ein einzelnes isoliertes
Datum in dem Speicher nur entschlüsselt werden kann, wenn die
gesamte Kette der sequentiellen Daten entschlüsselt wird.
Folglich ist kein direkter Zugriff auf Daten innerhalb der
10 CBC-Kette möglich. Das Durchgehen der Chiffre-Kette kostet
wiederum kostbare Rechenzeit und verbraucht unnötig viel
Strom, was insbesondere bei Smartcards von Nachteil ist, die
in batteriebetriebenen Geräten verwendet werden, wie z.B.
Handys, oder bei Chipkarten, bei denen die Kunden der Chip-
15 kartenherausgeber möglichst kurze Transaktionszeiten an den
Terminals verlangen.

Eine weitere Möglichkeit sicherzustellen, dass gleiche Klar-
texte, die sich an unterschiedlichen Speicherpositionen
20 befinden, in unterschiedliche Chiffretexte verschlüsselt
werden, besteht in der Generierung adressabhängiger Schlüssel
zur Verschlüsselung der Klartexte. Die Verwendung adressab-
hängiger Schlüssel macht sich den Umstand zunutze, dass einem
zu speichernden, zu verschlüsselnden Datum ein fester Spei-
cherplatz und somit eine fest zugeordnete Adresse zugeordnet
25 ist, und dass das verschlüsselte, gespeicherte Datum genau an
dieser fest zugeordneten Adresse gespeichert ist und bleibt,
bis es wieder anhand dieser Adresse ausgelesen wird. Aus
einem vorhandenen geheimen Generalschlüssel und der Adressin-
formation für eine Speicherposition bzw. eine individuell
30 adressierbare Einheit kann nun ein individueller Schlüssel
generiert werden, mit welchem dann das betreffende Datum bei
einem Schreibvorgang verschlüsselt und bei einem Lesevorgang
entschlüsselt werden kann. Die adressabhängige Generierung
35 hat den Nachteil, dass der Aufwand für die Schlüsselgenerie-
rung in etwa so groß ist, wie der Aufwand für die Verschlüs-
selung bzw. Entschlüsselung selbst, da für jeden adressierba-

ren Speicherplatz bzw. für jedes adressierbare Speicherwort eine Schlüsselgenerierung durchgeführt werden muss, die dafür sorgt, dass die Abbildung von Adresse auf zugehörige adress-abhängige Schlüssel für einen Unbefugten möglichst undurchsichtig ist. Schlüsselgenerierung auf Speicherwortgranularität zieht folglich ebenfalls ein hohes Maß an Performancereduzierung mit sich, was die Kundenfreundlichkeit bei beispielsweise Chipkarten verringern kann.

- 10 Eine Möglichkeit, das Sicherheitsdefizit durch Weglassen der Adressabhängigkeit bei der Verschlüsselung gespeicherter Speicherinhalte auszugeichen, bestünde darin, die Blockgrößen bei der Verschlüsselung zu vergrößern, da dies die Anzahl möglicher Klartexte für ein Chiffrat erhöht. Damit verbunden 15 ist jedoch ein erhöhter Aufwand auf Seiten der Verschlüsselungs- und Entschlüsselungshardware, der diese Option für Massenware wie Chipkarten untragbar macht.

Die Aufgabe der vorliegenden Erfindung besteht darin, ein 20 Verfahren und eine Vorrichtung zur Erzeugung individueller Schlüssel zu schaffen, die einen Zugriff auf einen Speicher auf der Basis dieser individuellen Schlüssel ermöglichen und dabei den Gesamtaufwand zum Zugriff zu verringern.

- 25 Diese Aufgabe wird durch eine Vorrichtung gemäß Anspruch 1 und ein Verfahren gemäß Anspruch 13 gelöst.

Die Erkenntnis der vorliegenden Erfindung besteht darin, dass die in vielen Systemen bereits vorhandene Gruppierung der 30 individuell adressierbaren Einheiten eines Speichers zu Gruppen bzw. Seiten verwendet werden kann, um die Komplexität der adressabhängigen Schlüsselgenerierung bei nur geringfügiger Verringerung an der Sicherheit deutlich zu verkleinern, wenn zunächst ein Seitenvorschlüssel auf der Basis einer 35 Seitenadresse berechnet wird und dann auf der Basis des Seitenvorschlüssels und der Wortadresse erst der individuelle Schlüssel ermittelt wird. Hierdurch kann die adressabhängige

Schlüsselgenerierung in einen kryptographisch anspruchsvollen relativ aufwendigen Prozess, der aber nur selten durchgeführt werden muss, nämlich die Seitenvorschlüsselberechnung, und in einen schnellen, nahezu aufwandslosen Schritt, der für jedes

5 Wort bzw. jede individuell adressierbare Einheit ausgeführt werden muss, nämlich die Ermittlung des individuellen Schlüssels auf der Basis des Seitenvorschlüssels und der Wortadresse, aufgespalten werden. Der Seitenvorschlüsselberechnungsvorgang kann also derart gewählt werden, dass der Vorgang bei

10 der Implementierung weniger Chipfläche und/oder mehr Verarbeitungslaufzeitdauer aufweist als die Implementierung der Ermittlung des individuellen Schlüssels. Hierdurch kann die Zugriffszeit auf den Speicher verringert werden, da die Seitenadresse ja für alle individuell adressierbaren Einheiten,

15 die zu einer Seite gehören, gleich ist und folglich nicht immer erneut berechnet werden muss. Vielmehr kann die Seitenadresse zwischengespeichert werden, um für diejenigen unter den nachfolgenden Zugriffen auf den Speicher zur Verfügung zu stehen, die sich auf adressierbare Einheiten in der

20 entsprechenden Speicherseite beziehen. Die Speicherung kann beispielsweise in einem Verdrängungsspeicher erfolgen, bei dem ein gewisser Verdrängungsmechanismus verwendet wird, um in demselben beispielsweise die Seitenvorschlüssel für diejenigen Seiten zeitweilig gespeichert bereitzustellen, auf die

25 bzw. deren inbegriffenen Einheiten mit hoher Wahrscheinlichkeit in kürze wieder zugegriffen wird. Dieser Speicher kann, wenn vorhanden, mit einem Cache- oder Daten-Cache-Speicher integriert bzw. kombiniert werden, der ja auf ähnlich Weise dazu da ist, aktuelle Daten für einen schnellen Zugriff bereitzustellen, ohne dass ein Zugriff auf einen langsameren

30 Hintergrundspeicher notwendig ist.

Bevorzugte Ausführungsbeispiele der vorliegenden Erfindung werden nachfolgend Bezug nehmend auf die beiliegenden Zeichnungen näher erläutert. Es zeigen:

35

- Fig. 1 ein Blockschaltbild eines Speichersystems, in welchem eine erfindungsgemäße Schlüsselerzeugung verwendet werden kann, gemäß einem Ausführungsbeispiel der vorliegenden Erfindung;
- 5
- Fig. 2a ein schematisches Blockschaltbild zur Veranschaulichung des Aufbaus und der Funktionsweise der Schlüsselerzeugungseinrichtung in dem Speichersystem von Fig. 1 gemäß einem Ausführungsbeispiel der vorliegenden Erfindung;
- 10
- Fig. 2b eine schematische Darstellung des Aufbaus einer Seite aus Wörtern gemäß einem Ausführungsbeispiel der vorliegenden Erfindung;
- 15
- Fig. 3 ein schematisches Blockschaltbild zur Veranschaulichung des Aufbaus und der Funktionsweise der Vorschlüsselberechnungseinrichtung in Fig. 2 gemäß einem Ausführungsbeispiel der vorliegenden Erfindung;
- 20
- Fig. 4 eine schematische Zeichnung zur Veranschaulichung des Aufbaus und der Funktionsweise der Einrichtung zur Ermittlung des individuellen Schlüssels aus den Seitenvorschlüsseln und der Wortadresse von Fig. 2 gemäß einem Ausführungsbeispiel der vorliegenden Erfindung;
- 25
- Fig. 5 eine schematische Zeichnung zur Veranschaulichung des Aufbaus und der Funktionsweise der Ermittlungseinrichtung von Fig. 2 gemäß einem weiteren Ausführungsbeispiel der vorliegenden Erfindung;
- 30
- Fig. 6 eine schematische Zeichnung zur Veranschaulichung des Aufbaus und der Funktionsweise der Ermittlungseinrichtung von Fig. 2 gemäß einem weiteren Ausführungsbeispiel der vorliegenden Erfindung;
- 35

Fig. 7 eine schematische Zeichnung zur Veranschaulichung des Aufbaus und der Funktionsweise der Ermittlungsseinrichtung von Fig. 2 gemäß einem weiteren Ausführungsbeispiel der vorliegenden Erfindung;

5

Fig. 8 ein Blockschaltbild eines für die Entschlüsselung zuständigen Teils einer Ver-/Entschlüsselungseinrichtung wie in Fig. 1 gemäß einem Ausführungsbeispiel der vorliegenden Erfindung;

10

Fig. 9 ein Blockschaltbild eines für die Verschlüsselung zuständigen Teils einer Ver-/Entschlüsselungsvorrichtung wie in Fig. 1 gemäß einem Ausführungsbeispiel der vorliegenden Erfindung; und

15

Fig. 10 ein Blockschaltbild einer Einrichtung zur Berechnung einer Rundenschlüsselfolge als dem individuellen Schlüssel aus dem Vorschlüssel gemäß einem weiteren Ausführungsbeispiel der vorliegenden Erfindung.

20

Bevor bezugnehmend auf die Zeichnungen die vorliegende Erfindung anhand von Ausführungsbeispielen näher erläutert wird, wird darauf hingewiesen, dass gleiche Elemente oder ähnliche Elemente in diesen Figuren mit gleichen oder ähnlichen Bezugssymbolen versehen sind, und dass eine wiederholte Beschreibung dieser Elemente vermieden wird.

25

Fig. 1 stellt ein System aus CPU 10, Speicher 12 und einer Speicherzugriffsvorrichtung 14 dar. Das System von Fig. 1 ist beispielsweise Teil eines Kryptocontrollers auf einer Chipkarte. In dem Speicher 12, der beispielsweise Teil eines physikalisch größeren Speichers sein kann, sind geheime Informationen gespeichert, wie z.B. ein Guthaben, ein Masterschlüssel eines chipkartenherausgebenden Institutes oder aber ein geheimer Code eines geheimen Kryptographiealgoritmus.

Die CPU 10 arbeitet ein Programm ab, das beispielsweise ebenfalls in dem Speicher 12 gespeichert sein kann, um vor Zugriffen unberechtigter geschützt zu sein. Einige Befehle in dem Programm erfordern, dass die CPU 10 Speicherinhalte in dem Speicher 12 lädt bzw. ausliest oder Speicherinhalte in dem Speicher 12 durch neue Informationen be- oder überschreibt.

Die Zugriffsvorrichtung 14 ist dazu vorgesehen, sicherzustellen, dass die geheimen Informationen in dem Speicher 12 immer in verschlüsselter Form gespeichert werden, und dass umgekehrt die verschlüsselten Speicherinhalte des Speichers 12 bei Ladevorgängen bzw. bei Auslesen derselben wieder entschlüsselt werden.

Die Zugriffsvorrichtung 14 umfasst eine Ver-/Entschlüsselungseinrichtung 16 sowie eine Schlüsselerzeugungseinrichtung 18. Die Ver-/Entschlüsselungseinrichtung 16 ist dazu vorgesehen, Daten von der CPU 10, die in dem Speicher 12 gespeichert werden sollen, vor ihrer Speicherung zu verschlüsseln, und von dem Speicher 12 ausgegebene, gespeicherte und verschlüsselte Daten vor ihrer Weiterleitung an die CPU 10 zu entschlüsseln. Die Ver-/Entschlüsselungseinrichtung 16 verwendet hierzu einen wortindividuellen Schlüssel, den sie von der Schlüsseleinrichtung 18 erhält.

Genauer ausgedrückt ist die CPU 10 über einen Adressbus 20 mit sowohl einem Adresseingang des Speichers 12 als auch einem Adresseingang der Schlüsselerzeugungseinrichtung 18 verbunden. Die Schlüsselerzeugungseinrichtung 18 gibt an ihrem Ausgang wortindividuelle Schlüssel für die Adressen auf dem Adressbus 20 aus, wobei der Ausgang der Schlüsselerzeugungseinrichtung 18 mit einem Schlüsseleingang der Ver-/Entschlüsselungseinrichtung 16 verbunden ist. Über einen Datenbus 22 ist die CPU 10 mit einem Daten-Ein/Ausgang des Speichers 12 verbunden. In den Datenbus 22 ist die Ver-/Entschlüsselungseinrichtung 16 geschaltet. Insbesondere ist

ein Dateneingang der Ver-/Entschlüsselungseinrichtung 16 mit einem Datenausgang der CPU 10 und ein weiterer Dateneingang der Ver-/Entschlüsselungseinrichtung 16 mit einem Datenausgang des Speichers 12 verbunden, während ein Datenausgang der 5 Ver-/Entschlüsselungseinrichtung 16 mit einem Dateneingang des Speichers 12 und ein weiterer Datenausgang der Ver-/Entschlüsselungseinrichtung 16 mit einem Dateneingang der CPU 10 verbunden ist. Die Ver-/Entschlüsselungseinrichtung 16 bildet also eine Schnittstelle zwischen CPU 10 und Speicher 12 und 10 sorgt dafür, dass auf dem Datenbus 22 zwischen CPU 10 und Ver-/Entschlüsselungseinrichtung 16 die Daten in entschlüsselter Form, d.h. im Klartext, und in dem Teil des Datenbusses 22 zwischen dem Speicher 12 und der Ver-/Entschlüsselungseinrichtung 16 nur in verschlüsselter Form, d.h. als 15 Chiffrat, vorkommen.

Nachdem im Vorhergehenden der Aufbau des Systems von Fig. 1 beschrieben worden ist, wird im Folgenden dessen Funktionsweise beschrieben. Bei der nachfolgenden Beschreibung der 20 Funktionsweise der CPU 10 wird angenommen, dass sich in dem Speicher 12 bereits verschlüsselte Speicherinhalte befinden. Der Speicher 12 ist in kleinste individuell adressierbare Einheiten, im Folgenden als Wörter bezeichnet, gegliedert. Jedem Wort des Speichers 12 ist eine eindeutige Adresse 25 zugeordnet. Wird nun die CPU 10 bei Abarbeitung eines Programms angewiesen einen Ladevorgang durchzuführen, d.h. den verschlüsselten Speicherinhalt an einem individuell adressierbaren Element bzw. ein Wort aus dem Speicher 12 zu laden, so gibt die CPU 10 auf dem Adressbus 20 die entsprechende 30 eindeutige Adresse aus. Der Speicher 12 verwendet die Adresse um auf die entsprechende physikalische Speicherposition zuzugreifen und das dort gespeicherte verschlüsselte Wort auszulesen und auf den Datenbus 22 an die Ver-/Entschlüsselungseinrichtung 16 auszugeben.

35

Die Schlüsselerzeugungseinrichtung 18 empfängt die von der CPU 10 ausgegebene Adresse ebenfalls. Wie es im Folgenden

noch beschrieben werden wird, ermittelt die Schlüsselerzeugungseinrichtung 18 aus der Adresse auf dem Adressbus 20 einen wortindividuellen Schlüssel, den die Ver-/Entschlüsselungseinrichtung 16 benötigt, um den verschlüsselten ausgelesenen Speicherinhalt von dem Speicher 12 entschlüsseln zu können. Die Schlüsselerzeugungseinrichtung 18 gibt den wortindividuellen Schlüssel deshalb an die Ver-/Entschlüsselungseinrichtung 16 aus. Diese entschlüsselt auf der Basis des wortindividuellen Schlüssels den von dem Speicher 12 erhaltenen Speicherinhalt und gibt diesen Speicherinhalt im Klartext an die CPU 10 aus, die den nun entschlüsselten Speicherinhalt je nach Befehl im Klartext verarbeitet, wie z.B. in ein internes Register lädt oder dergleichen.

In dem Fall, dass das auszuführende Programm in der auszuführenden Befehlszeile einen Schreibvorgang anzeigt, gibt die CPU 10 auf dem Adressbus 20 die Adresse aus, die das Wort angibt, in welchem ein durch den Schreibbefehl spezifiziertes Datum gespeichert werden soll. Das zu speichernde Datum gibt die CPU 10 auf dem Datenbus 22 an die Ver-/Entschlüsselungseinrichtung 16 aus. Wie zuvor bei dem Ladevorgang erzeugt die Schlüsselerzeugungseinrichtung 18 aus der Adresse auf dem Adressbus 20 einen wortindividuellen Schlüssel und gibt denselben an die Ver-/Entschlüsselungseinrichtung 16 aus. Diese verwendet den wortindividuellen Schlüssel zur Verschlüsselung des zu speichernden Datums und gibt das Chiffrat an den Speicher 12 aus. Der Speicher 12 speichert das empfangene Chiffrat an der durch die Adresse auf dem Adressbus 20 angegebenen Stelle in dem entsprechenden Wort ab.

Bei der bisherigen Beschreibung wurde noch nicht genau auf die Funktionsweise der Schlüsselerzeugungseinrichtung 18 eingegangen. Wie es die im Folgenden beschriebenen Ausführungsbeispiele für die Schlüsselerzeugungseinrichtung 18 zeigen werden, ist die Schlüsselerzeugungseinrichtung 18 derart ausgebildet, dass sie in der Lage ist, wortindividuelle Schlüssel zur Verschlüsselung der Speicherinhalte bzw. Wörter

in dem Speicher 12 zu erzeugen, ohne aber dabei jedes Mal einen in etwa gleich aufwendigen Schlüsselerzeugungsprozess durchzuführen. Dies wird erzielt, indem mehrere Wörter, die ja die kleinsten adressierbaren Dateneinheiten des Speichers 5 12 darstellen, jeweils zu einer Seite zusammengefasst werden, und dass für eine Seite nur die Seitenadresse, die die Seite unter den Seiten des Speichers 12 spezifiziert, auf komplizierte und aufwendige und damit sichere Weise zur Erzeugung eines Seitenvorschlüssels herangezogen wird, während für die 10 Wörter innerhalb dieser Seite die wortindividuellen Schlüssel auf einfache, weniger kompliziertere Weise auf der Basis des Seitenvorschlüssels erzeugt werden. Lädt dann, wie es häufig vorkommt, die CPU 10 nacheinander die Wörter einer Seite aus, dann ist es nur beim ersten Wort dieser Seite notwendig, die 15 aufwendige Vorschlüsselgenerierung durchzuführen, während für die restlichen Wörter der Seite dann nur noch die weniger aufwendige Herleitung der wortindividuellen Schlüssel auf der Basis dieses Vorschlüssels notwendig ist.

20 Um die Aufgliederung des Speichers 12 in Seiten und Wörter näher zu veranschaulichen, wird zunächst auf Fig. 2a und 2b Bezug genommen. Fig. 2a zeigt in ihrer oberen Hälfte schematisch als Rechteck dargestellt einen Ausschnitt aus dem Speicher 12. Der in Fig. 2a dargestellte Ausschnitt umfasst exemplarisch 15 Seiten 12a. Jede Seite umfasst wiederum 16 Wörter 12b, wobei dies der Übersichtlichkeit halber in Fig. 2a jedoch nur für eine Seite 12a dargestellt ist. Die Wörter 12b des Speichers 12 sind die kleinsten adressierbaren Daten- 25 einheiten des Speichers 12. Das bedeutet anders ausgedrückt, dass jedem Wort 12b eine eindeutige Adresse zugeordnet ist, durch die es mit dem Speicher 12 verbundenen Einheiten, wie z.B. der CPU 10 von Fig. 1, möglich ist, auf die Speicherinhalte der einzelnen Wörter 12b zuzugreifen. Es wird darauf hingewiesen, dass die räumliche Anordnung der Wörter 12b, wie 30 sie in Fig. 2a dargestellt ist, freilich nur exemplarisch ist, und dass der Speicher 12 auch mehrere übereinander angeordnete Speicherebenen aufweisen kann, und dass die einzelnen 35

Wörter 12b auch anders zu Seiten zusammengefasst sein können oder anders angeordnet sein können als in Zeilen und Spalten, wie in Fig. 2a gezeigt.

- 5 Um nun die Darstellung der nachfolgenden Beschreibung zu vereinfachen, wird im Folgenden davon ausgegangen, dass der Speicher 12 2^{20} Wörter umfasse. Jedes Wort bestehe aus $32 = 2^5$ Bits. Die Speichergröße des Speichers 12 beträgt also nach diesem lediglich illustrativen Beispiel 2^{25} Bit = $2^{17} \times 2^8$ Bit
10 = 128 Kilobyte. Die Einteilung in Seiten wird exemplarisch derart vorgenommen, dass alle Wörter mit Adressen, deren 16 höchstwertige Bits (MSB; MSB = Most Significant Bits) gleich sind, zu einer Seite gehören bzw. zu einer Seite zusammengefasst werden. Die Wortadressen von Wörtern in einer Seite unterscheiden sich folglich lediglich in den verbleibenden 15 vier niedrigstwertigen Bits (LSB; LSB = Least Significant Bits) der Zwanzig-Bit-Wortadresse. Der Speicher 12 enthält folglich 2^{16} Seiten.
- 20 In Fig. 2a ist bei 30 noch exemplarisch der Aufbau einer eindeutigen Adresse eines Worts 12b gezeigt. Wie bereits erwähnt, wird im Folgenden davon ausgegangen, dass die eindeutige Adresse 30 20 Bit lang ist. Der höherwertigere Teil der eindeutigen Adresse 30 bestehend aus den 16 MSBs und wird 25 als der Seitenadressteil 30a der eindeutigen Adresse bezeichnet. Der Seitenadressteil 30a enthält die sog. Seitenadresse. Der niedrigwertigere Teil 30b der eindeutigen Adresse wird aus den vier LSBs gebildet und stellt den Wortadressteil 30b dar. Der Wortadressteil 30b gibt an, welchem der Wörter in 30 der durch den Seitenadressteil angegebenen Seite 12a die eindeutige Adresse 30 zugeordnet ist.

- Dies wird in Fig. 2b näher veranschaulicht, die für eine Seite 12b die Aufteilung in die 16 Wörter 12a dieser Seite 35 12b veranschaulicht, und in der die Wörter durchnummieriert und mit einer zugehörigen 4-Bit-Wortadresse bzw. mit einem 4-Bit-Versatzwert versehen sind, der für das jeweilige Wort in

dem Wortadressteil 30b enthalten sein muss. Durch die Wortadresse bzw. den Versatzwert in dem Wortadressteil 30b wird folglich jedem Wort einer Seite ein fester Platz innerhalb der Seite zugewiesen.

5

Bezugnehmend auf Fig. 2a wird nun im Folgenden die Schlüsselerzeugungseinrichtung 18 gemäß einem Ausführungsbeispiel der vorliegenden Erfindung näher beschrieben. Die Schlüsselerzeugungseinrichtung 18 umfasst eine Seitenwechselfeststellungs-
10 einrichtung 32, eine Vorschlüsselberechnungseinrichtung 34 und eine Einrichtung 36 zur Ermittlung des individuellen Schlüssels. Ferner umfasst die Schlüsselerzeugungseinrichtung 18 einen Speicher, wie z.B. einen flüchtigen Speicher, 38 zur vorübergehenden Speicherung von dem oder den zuletzt verwen-
15 deten Seitenvorschlüssel(n), auf deren Funktion und Erzeugung noch im Folgenden näher eingegangen wird.

Die Seitenwechselfeststellungseinrichtung 32 ist dazu vorgesehen, den Seitenadressteil 30a der eindeutigen Adresse 30
20 auf dem Adressbus 20 zu empfangen und daraufhin zu überprüfen, ob dieselbe eine Seite betrifft, für die sich bereits ein Seitenvorschlüssel in dem Zwischenspeicher 38 befindet. Ist dies der Fall, ist die Seitenwechselfeststellungseinrich-
tung 32 in der Lage, anhand der dort gespeicherten Seitenad-
25 resse auf den Zwischenspeicher 38 zuzugreifen, welcher dar- aufhin den zwischengespeicherten Seitenvorschlüssel für die durch die Seitenadresse angegebene Seite an einen Seiten- vorschlüsseleingang der Einrichtung 36 weiterleitet.

30 Stellt die Seitenwechselfeststellungseinrichtung 32 fest, dass für die durch die Seitenadresse angezeigte Seite noch kein Seitenvorschlüssel vorliegt, gibt dieselbe die Seitenad- resse an einen Seitenadresseingang der Vorschlüsselberech- nungseinrichtung 34 weiter. Die Vorschlüsselberechnungsein-
35 rrichtung 34 berechnet aus der Seitenadresse einen Seiten- vorschlüssel für diese Seite und gibt den berechneten Seiten- vorschlüssel an den Seitenvorschlüsseleingang der Einrichtung

36 weiter. Ferner gibt sie denselben unter Verdrängung eines bereits gespeicherten Seitenvorschlüssels zur vorübergehenden Speicherung an den Speicher 38 aus.

- 5 Der Wortadressteil 30b der auf dem Adressbus 20 vorliegenden eindeutigen Adresse 30 wird an einen Wortadresseingang der Einrichtung 36 zur Ermittlung des individuellen Schlüssels weitergeleitet. Die Einrichtung 36 ermittelt aus eingegangener Wortadresse bzw. eingegangenem Versatzwert und eingegangenem Seitenvorschlüssel einen wortindividuellen Schlüssel für das Wort, auf die die eindeutige Adresse 30 zeigt. Die Einrichtung 36 gibt diesen wortindividuellen Schlüssel an ihrem Ausgang aus, der gleichzeitig auch den Ausgang der Schlüsselerzeugungseinrichtung 18 darstellt, der, wie in Fig. 15 gezeigt, mit dem Schlüsseleingang der Ver-/Entschlüsselungseinrichtung 16 verbunden ist.

Nachdem im Vorhergehenden der Aufbau sowie die Funktionsweise der einzelnen Komponenten der Schlüsselerzeugungseinrichtung 18 beschrieben worden ist, wird im Folgenden die Funktionsweise beschrieben. Zu diesem Zweck wird zunächst davon ausgegangen, dass noch zu keiner Seite 12a des Speichers 12 ein Seitenvorschlüssel erzeugt worden ist, also auch noch keiner im Speicher 38 gespeichert ist.

- 25 Bei Eingang einer eindeutigen Adresse 30 schlägt die Seitenwechselfeststellungseinrichtung 32 zunächst im Speicher 38 nach, ob für die in dem Seitenadressteil 30a enthaltene Seitenadresse der Seite, in der sich das Wort befindet, auf die die eindeutige Adresse 30 zeigt, wie es durch einen Pfeil 40 angedeutet ist, ein Seitenvorschlüssel vorhanden ist. Das Nachschlagen findet unter Verwendung der Seitenadresse als Index statt, wobei in dem Speicher 38 eine Tabelle von Seitenadress/Seitenvorschlüssel-Paaren bereitgestellt ist. Der Speicher 38 ist somit ein inhaltsadressierbarer Speicher, der für ankommende Seitenadressen nachschlägt, ob er ein Seitenadress/Seitenvorschlüssel-Paar mit dieser Seitenadresse hat

und den entsprechenden Schlüssel ausgibt, falls dies der Fall ist. Dies soll jedoch vorliegend gemäß obiger Annahme zunächst nicht der Fall sein. Der Speicher 38 teilt den Fehlschlag (miss) folglich der Seitenwechselfeststellungseinrichtung 32 mit, die ihrerseits wiederum die Seitenadresse an die Vorschlüsselberechnungseinrichtung 34 weiterleitet.

- Die Vorschlüsselberechnungseinrichtung 34 berechnet auf der Basis der Seitenadresse einen Seitenvorschlüssel. Diese Berechnung umfasst gemäß dem bezugnehmend auf Fig. 3 näher beschriebenen Ausführungsbeispiel wie die Verschlüsselungseinrichtung 34 beispielsweise eine Verschlüsselung der Seitenadresse unter Verwendung eines Generalschlüssels, so dass die Korrelation zwischen Seitenvorschlüsseln, die die Vorschlüsselberechnungseinrichtung 34 aus den Seitenadressen berechnet, zu den Seitenadressen möglichst kompliziert ist. Die Vorschlüsselberechnungseinrichtung 34 kann beispielsweise ganz allgemein die Seitenadresse an ihrem Seitenadresseingang gemäß einer nichtlinearen Abbildung auf die Seitenvorschlüssel abbilden. Diese nichtlineare Abbildung kann jedweder Abbildung der 16-Bit-SeitenAdresse auf einen M-Bit-Seitenvorschlüssel sein, wobei $m \geq 16$ sein muss, damit wirklich für jede Seite ein unterschiedlicher Seitenvorschlüssel generiert wird, m aber auch kleiner 16 sein kann, wenn dies nicht erforderlich ist. Die Vorschlüsselberechnungseinrichtung 34 kann auch eine Einwegfunktion oder eine asymmetrische Verschlüsselung verwenden, um die Seitenadresse auf den Seitenvorschlüssel abzubilden.
- Der den Versatzwert enthaltende Wortadressteil 30b der auf dem Datenbus 20 vorliegenden eindeutigen Adresse 30 und der soeben von der Vorschlüsselberechnungseinrichtung 34 berechnete Seitenvorschlüssel wird von der Einrichtung 36 dazu verwendet, den wortindividuellen Schlüssel zu berechnen. Wie es später bezugnehmend auf die Fig. 4 - 7 noch näher erläutert werden wird, kann die Einrichtung zur Ermittlung des individuellen Schlüssels 36 beispielsweise den Versatzwert

bzw. die Wortadresse an ihren Wortadresseingang dazu verwenden, den Seitenvorschlüssel zu modifizieren, und zwar unter Verwendung einfacher Rechenoperationen, wie z.B. XOR- oder NXOR-Operationen oder dergleichen. Vorzugsweise sollten die

5 Rechenoperationen, die die Einrichtung 36 zur Ermittlung des individuellen Schlüssels verwendet, geringer an Zahl und weniger zeitaufwendig sein als die Vorschlüsselberechnungseinrichtung 34 zur Berechnung des Seitenvorschlüssels verwendet. Ferner wird es bevorzugt, dass die Einrichtung 36 zur

10 Ermittlung des individuellen Schlüssels mit weniger Chipfläche implementiert ist und/oder die Implementierung der Einrichtung 36 zur Ermittlung des individuellen Schlüssels eine geringere Berechnungszeitdauer aufweist als die Vorschlüsselberechnungseinrichtung 34.

15

Der wortindividuelle Schlüssel, wie er von der Einrichtung 36 ermittelt worden ist, wird dann an die Ver-/Entschlüsselungseinrichtung 16 weitergeleitet.

20 Um nun zu verhindern, dass für jede eindeutige Adresse 30, die auf dem Adressbus 20 ausgegeben wird, die komplizierte Vorschlüsselberechnung erneut durchgeführt werden muss, speichert die Vorschlüsselberechnungseinrichtung 34 bei jeder Berechnung den soeben berechneten Seitenvorschlüssel in dem

25 Zwischenspeicher 38. Der Zwischenspeicher 38 wird beispielsweise nach dem FIFO-Prinzip (FIFO = zuerst hinein zuerst hinaus) verwaltet, so dass bei Speicherung eines neuen Seitenvorschlüssels durch die Vorschlüsselberechnungseinrichtung 34 derjenige Seitenvorschlüssel verdrängt bzw. überschrieben

30 wird, der zuerst hineingeschrieben worden ist. Freilich sind auch andere Aktualisierungs- bzw. Verdrängungsprinzipien anwendbar, wie z.B. das LRU- (Least Recently Used) Prinzip, bei dem der Seitenvorschlüssel ausgelagert wird, auf den die längste Zeit nicht mehr durch die Seitenwechselfeststellungseinrichtung 32 zugegriffen worden ist, oder das LFU- (Least Frequently Used) Prinzip, bei dem der Seitenvorschlüssel

ausgelagert wird, der die geringste Anzahl von Zugriffen durch die Seitenwechselfeststellungseinrichtung 32 aufweist.

Wenn nun eine folgende eindeutige Adresse 30, die auf dem Adressbus 20 danach ausgegeben wird, einen Seitenadressteil 30a aufweist, in der eine Seitenadresse enthalten ist, die eine Seite anzeigt, für die in dem Zwischenspeicher 38 noch ein Seitenvorschlüssel zwischengespeichert ist, zeigt dies der Speicher 38 der Seitenwechselfeststellungseinrichtung 32 auf Erhalt der Seitenadresse als Index hin durch ein Treffer-(hit) Signal an. Die Seitenwechselfeststellungseinrichtung 32 leitet die Seitenadresse hierauf nicht an den Seitenadress-eingang der Verschlüsselungsberechnungseinrichtung 34 weiter, sondern unter Umgehung der letztgenannten wird der der interessierenden Seitenadresse zugeordnete und bereits früher berechnete Seitenvorschlüssel von dem Speicher 38 an den Seitenvorschlüsseleingang der Einrichtung 36 ausgegeben. Aufgrund der Umgehung der Vorschlüsselberechnungseinrichtung 34 im Fall eines Treffers muss in diesem Fall also keine komplizierte und zeitraubende Vorschlüsselberechnung durchgeführt werden. Lediglich ein Nachschlagvorgang in dem Zwischenspeicher 38 ist notwendig, um den Seitenvorschlüssel für die neue eindeutige Adresse 30 auf dem Adressbus 20 zu ermitteln. Hierbei können zwischen der eindeutigen Adresse 30, auf deren Ausgabe auf dem Adressbus 20 hin der Seitenvorschlüssel tatsächlich durch die Vorschlüsselberechnungseinrichtung 34 berechnet und in den Zwischenspeicher 38 eingetragen worden ist, und der eindeutigen Adresse 30, auf welche hin der Seitenvorschlüssel aus dem Speicher 38 abgerufen wird, auf dem Adressbus 20 zwischenzeitlich schon eine oder mehr eindeutige Adressen 30 ausgegeben worden sein. Das bedeutet, dass bei geeigneter, an die jeweilige Anwendung des Systems von Fig. 1 angepasster Verdrängungsstrategie des Speichers 38 eventuell über lange Zeitdauern hinweg bzw. über mehrere eindeutige Adressen 30 auf dem Adressbus 20 hinweg, keine Vorschlüsselberechnung durchgeführt werden muss. Lediglich die unkomplizierte Modifizierung des von dem Zwischenspeicher

38 abgerufenen Seitenvorschlüssels durch die Wortadresse bzw. den Versatzwert in der Einrichtung 36 muss durchgeführt werden, um zu dem wortindividuellen Schlüssel für das Wort zu gelangen, dem die eindeutige Adresse 30 auf dem Adressbus 20 5 zugeordnet ist.

Der Zwischenspeicher 38 kann in einem der CPU 10 zugeordneten Cash-Speicher (nicht gezeigt in Fig. 1) zu einem erweiterten Cache integriert sein. Die Cache enthielt dann beispielsweise für einen schnelleren Datenzugriff die Inhalte kürzlich verwendeter Seiten, d.h. Seiten auf die kürzlich zugegriffen wurde. Die Inhalte wären dann verschlüsselt gespeichert. Zu jedem Seiteninhalt wäre in dem Cache die zugehörige Seitenadresse und der Seitenvorschlüssel gespeichert. Dieser erweiterte Cache enthielt also eine Tabelle von Triplets aus Seitenadresse, Seitenvorschlüssel und Seitenchiffrat. Die Verdrängungsstrategie dieses erweiterten Caches könnte eine der vorhergehenden genannten sein. In dem Fall eines Fehlschlags wäre die Funktionsweise die gleiche, wie oben beschrieben, mit dem Unterschied, dass nun auch der verschlüsselte Inhalt der aktuellen Seite unter Verdrängung einer anderen Seite in dem Cache gespeichert wird. In dem Fall eines Treffers würde neben der Ausgabe des Seitenvorschlüssels an die Einrichtung 36 ferner die Ausgabe des verschlüsselten Worts, auf den die eindeutige Adresse 30 zeigt, aus dem Chiffrat der interessierenden Seite an den Dateneingang der Ver/Entschlüsselungseinrichtung 16 (Fig. 1) ausgegebene werden, der mit dem Speicher 12 verbunden ist, wodurch der langsamere Zugriff auf den Speicher 12 umgangen würde. Freilich könnte der Cache-Speicher auch wortweise organisiert sein mit Tripeln aus Wortadresse, Wortchiffrat und Seitenvorschlüssel. In dem Fall eines Daten-Caches mit unverschlüsselten Daten wäre dieser in der Klartext-Domäne zwischen CPU und Vorrichtung 14 angeordnet (Fig. 1) und eine Kombination des Daten-Caches mit der Speicher 38 wäre unnötig. Ferner könnte aber anstelle des Cache-Speichers 38 auch ein einfacher Speicher für die Speicherung nur des zuletzt berechneten

Seitenvorschlüssels verwendet werden, dessen Inhalt bei einem nachfolgenden Zugriffsvorgang ausgelesen und unter Umgehung der Einrichtung 34 verwendet wird, wenn die nächste Seitenadresse, auf den sich dieser Zugriffsvorgang bezieht, die selbe 5 wie die ist, auf den sich der vorhergehende Speichervorgang bezog, bei dem der zuletzt berechnete Speichervorschlüssel berechnet und gespeichert wurde.

Bezugnehmend auf Fig. 3 wird nun im Folgenden ein Ausführungsbeispiel für die Vorschlüsselberechnungseinrichtung 34 beschrieben. Die Vorschlüsselberechnungseinrichtung 34 von Fig. 3 umfasst einen Seitenadresseingang 50 und einen Vorschlüsselausgang 52, sowie eine Expansionseinrichtung 54 und ein Blockchiffremodul 56. Ein Dateneingang der Expansionseinrichtung 54 ist mit dem Seitenadresseingang 50 verbunden, um die 16-Bit-Seitenadresse zu erhalten. Die Expansionseinrichtung 54 erzeugt auf Basis der 16-Bit-Seitenadresse einen 64-Bit-Datenblock 58, indem er jedes Bit der Seitenadresse auf vier Bitpositionen des 64-Bit-Datenblocks 58 verteilt. Genauer ausgedrückt, schreibt die Expansionseinrichtung 54 die Seitenadresse in die Bits 0 - 15, 16 - 31, 32 - 47 und 48 - 63 des 64-Bit-Datenblocks 58, wie es auch in Fig. 3 angedeutet ist. Den so gebildeten 64-Bit-Datenblock 58 gibt die Expansionseinrichtung 54 an einen Dateneingang des Blockchiffremoduls 56 aus. Das Blockchiffremodul 56 umfasst neben dem Dateneingang ferner einen Schlüsseleingang, an dem dieselbe einen 64-Bit-Generalschlüssel erhält. Der 64-Bit-Generalschlüssel ist fest und muss vor einem unautorisierten Zugriff durch Dritte durch geeignete Maßnahmen geschützt 20 werden. Das Blockchiffremodul 56 bewirkt eine nichtlineare Abbildung, die von dem Generalschlüssel abhängt, und ist beispielsweise ein DES-Modul oder ein AES-Modul. Basierend auf dem Generalschlüssel chiffriert bzw. verschlüsselt das Blockchiffremodul 56 den 64-Bit-Datenblock 58, um einen 25 verschlüsselten 64-Bit-Datenblock zu erhalten und diesen an den Datenausgang 52 auszugeben. Dieser verschlüsselte 64-Bit-Datenblock stellt gemäß diesem Ausführungsbeispiel den Sei- 30 35

tenvorschlüssel dar, basierend auf welchem, wie in Fig. 2a beschrieben, die wortindividuellen Schlüssel der Wörter der zugehörigen Seite berechnet werden, auf die die Seitenadresse am Eingang 50 zeigt.

5

Bezugnehmend auf die folgenden Fig. 4 - 7 werden nun Ausführungsbeispiele für die Einrichtung 36 zur Ermittlung des individuellen Schlüssels aus dem Seitenvorschlüssel und der Wortadresse von Fig. 2a beschrieben.

10

Gemäß dem Ausführungsbeispiel von Fig. 4 wird die 4-Bit-Wortadresse durch XOR-Operationen mehrmals auf den Seiten-schlüssel aufaddiert. Hierzu umfasst die Einrichtung 36 16 4-Bit-XOR-Verknüpfungseinrichtungen 60a, 60b, 60c, 60d und 60e.

15

Jede XOR-Verknüpfungseinrichtung umfasst zwei 4-Bit-Dateneingänge und einen 4-Bit-Datenausgang. An dem ersten der beiden 4-Bit-Dateneingänge empfangen die XOR-Verknüpfungseinrichtungen 60a - 60e die 4-Bit-Wortadresse. An den zweiten Dateneingängen empfangen die XOR-

20

Verknüpfungseinrichtungen 60a - 60e unterschiedliche vier Bits des 64-Bit-Vorschlüssels, der beispielsweise auf die in Fig. 3 gezeigte Weise erzeugt worden ist. Genauer ausge- drückt, empfängt jede XOR-Verknüpfungseinrichtung 60a - 60e unterschiedliche vier aufeinanderfolgende Bits des Seiten-

25

vorschlüssels, nämlich die XOR-Verknüpfungseinrichtung 60a die Bits der Bitpositionen 63 - 60, die XOR-Verknüpfungseinrichtung 60b die Bits der Bitpositionen 59 - 56, die XOR-Verknüpfungseinrichtung 60c die Bits der Bitposi- tionen 55 ... 52 usw. des Vorschlüssels, wie es in Fig. 4

30

angezeigt ist. Die XOR-Verknüpfungseinrichtungen 60a - 60e verknüpfen bitweise die 4-Bit-Wortadresse mit den jeweiligen vier Bits aus dem 64-Bit-Vorschlüssel.

35

Gegeben sei beispielsweise der Seitenschlüssel als der Vektor $\bar{K}_{\text{Seite}} = (k_0 \ k_1 \ k_2 \dots \ k_{62} \ k_{63})$, wobei k_i , für $i = 0 \dots 63$, der Bitwert des Seitenvorschlüssels an der Bitposition i sei, und die Wortadresse durch den Vektor $(w_0 \ w_1 \ w_2 \ w_3)$, wobei w_j mit j

- = 0 ... 3 das Bit der Wortadresse an der Bitposition j sei. Die bitweisen XOR-Verknüpfungsergebnisse der XOR-Gatter 60a - 60i werden dann derart zu einem wortindividuellen Schlüssel mit 64 Bit zusammengefasst, dass sich der wortindividuelle
- 5 Schlüssel zu einem Vektor \bar{K}_{Wort} ergibt, mit $\bar{K}_{\text{Wort}} = (k_0 \oplus w_0, k_1 \oplus w_1, k_2 \oplus w_2, k_3 \oplus w_3, k_4 \oplus w_0, k_5 \oplus w_1, k_6 \oplus w_2, k_7 \oplus w_3, \dots, k_{60} \oplus w_0, k_{61} \oplus w_1, k_{62} \oplus w_2, k_{63} \oplus w_3)$, wobei \oplus eine XOR-Verknüpfung anzeigt.
- 10 Gemäß dem Ausführungsbeispiel von Fig. 5 umfasst die Einrichtung 36 zur Ermittlung des individuellen Schlüssels eine Nachschlagetabelle 70, die für jede mögliche 4-Bit-Wortadresse einen unterschiedlichen 64-Bit-Maskierungsvektor bereitstellt, sowie eine 64-Bit-XOR-Verknüpfungseinrichtung 72.
- 15 Gemäß diesem Ausführungsbeispiel wird mit der Wortadresse als Index durch eine mit 74 veranschaulichte Nachschlageinrichtung beispielsweise auf die Nachschlagetabelle 70 zugegriffen, um auf den Maskierungsvektor zuzugreifen, der der Wortadresse zugeordnet ist, die der Einrichtung 36 von dem Wortadressteil
- 20 der aktuell auf dem Adressbus vorliegenden eindeutigen Adresse zugeführt wird. Die Nachschlagetabelle 70 gibt den indexierten Maskierungsvektor an einen ersten Dateneingang der XOR-Verknüpfungseinrichtung 72 aus. Ein zweiter 64-Bit-Dateneingang der XOR-Verknüpfungseinrichtung 72 erhält den
- 25 64-Bit-Seitenvorschlüssel. Ein 64-Bit-Datenausgang der XOR-Verknüpfungseinrichtung 72 stellt gleichzeitig den Ausgang der Einrichtung 36 von Fig. 5 dar. An diesem wird der wortindividuelle 64-Bit-Schlüssel ausgegeben.
- 30 Sei der Seitenvorschlüssel beispielsweise durch den bezugnehmend auf Fig. 4 definierten Vektor \bar{K}_{Seite} gegeben, so gibt die XOR-Verknüpfungseinrichtung 72 an ihren Datenausgang $(k_0 \oplus m_0, k_1 \oplus m_1, \dots, k_{63} \oplus m_{63})$ als wortindividuellen Schlüssel \bar{K}_{Wort} aus, wenn der indexierte Maskierungsvektor $(m_0, m_1, \dots, m_{63})$ ist. Anders ausgedrückt, wird gemäß dem Ausführungsbeispiel von Fig. 5 für jede Wortadresse ein Maskierungsvektor bereitgestellt, der die selbe Bitlänge wie der Seiten-

vorschlüssel aufweist, wobei über eine Nachschlagetabelle für die interessierende Wortadresse der ihr zugeordnete Maskierungsvektor angesteuert und dieser Maskierungsvektor dann auf den Seitenvorschlüssel durch bzw. per XOR-Verknüpfung aufaddiert wird, wobei das Ergebnis der wortindividuelle 64-Bit-Schlüssel ist.

Gemäß dem Ausführungsbeispiel von Fig. 6 umfasst die Einrichtung 36 zur Ermittlung des wortindividuellen Schlüssels eine Nachschlagetabelle 80, in der vordefinierte unterschiedliche Permutationsvorschriften, nämlich eine unterschiedliche Permutationsvorschrift für jeden möglichen Wert, den die Wortadresse annehmen kann, und eine steuerbare Permutationseinrichtung 82. Mit dem der Einrichtung 36 zugeführten Wortadresse von dem Wortadressteil der aktuell auf dem Adressbus vorliegenden eindeutigen Adresse als Index greift ein mit 84 versinnbildlichte Nachschlageeinrichtung auf die Nachschlagetabelle 80 zu, die daraufhin die dieser Wortadresse zugeordnete Permutationsvorschrift an die steuerbare Permutationseinrichtung 32 weiterleitet. Die Permutationsvorschriften, die in der Nachschlagetabelle 80 für jeden möglichen Wert der Wortadressen gespeichert sind, sind beispielsweise 64-Bit-Vektoren mit 6-Bit-Koeffizienten, wobei der erste Koeffizient angibt, an welcher Stelle gemäß der Permutationsvorschrift das niedrigstwertige Bit an dem 64-Bit-Dateneingang der steuerbaren Permutationseinrichtung 82 verschoben wird, der zweite Koeffizient angibt, wohin das nächste höherwertige Bit am 64-Bit-Dateneingang der steuerbaren Permutationseinrichtung 82 verschoben werden soll, usw. Der Dateneingang der steuerbaren Permutationseinrichtung 82 empfängt den Seitenvorschlüssel. Am 64-Bit-Datenausgang der steuerbaren Permutationseinrichtung 82 wird der wortindividuelle 64-Bit-Schlüssel ausgegeben, der sich gemäß der indexierten Permutationsvorschrift lediglich dadurch von dem Seitenvorschlüssel unterscheidet, dass die einzelnen Bits innerhalb der 64 Bitpositionen verschoben bzw. untereinander vertauscht sind.

Sei der Seitenvorschlüssel beispielsweise durch den bezugnehmend auf Fig. 4 definierten Vektor \bar{K}_{Seite} gegeben, so gibt die Permutationseinrichtung 72 an ihren Datenausgang ($k_{P(0)}$, $k_{P(1)}$, ..., $k_{P(63)}$) als wortindividuellen Schlüssel \bar{K}_{Wort} aus,

5 wenn der ausgewählte Permutationsvektor ($P(0)$, $P(1)$, ..., $P(63)$) ist.

Anders ausgedrückt, steht gemäß dem Ausführungsbeispiel von Fig. 6 für jede Wortadresse eine bestimmte Permutation bereit. Zur Generierung des wortindividuellen Schlüssels aus

10 dem Seitenvorschlüssel lässt man die der Wortadresse zugeordnete Permutation auf die Bits des Seitenvorschlüssels wirken, wobei das Ergebnis dann der wortindividuelle Schlüssel ist.

15 Im Bezug auf die vorhergehenden bezugnehmend auf die Fig. 4 - 6 beschriebenen Ausführungsbeispiele für die Einrichtung 36 zur Ermittlung des wortindividuellen Schlüssels gilt, dass dieselben ohne weiteres mit dem Ausführungsbeispiel von Fig. 3 für die Vorschlüsselberechnungseinrichtung 34 kombinierbar

20 sind, da dieselben als Seitenvorschlüssel einen 64-Bit-Seitenvorschlüssel erwarten bzw. verlangen. Wie aber bereits erwähnt, ist es ferner möglich, die Vorschlüsselberechnungseinrichtung 34 anders auszuführen. So könnte es möglich sein, dass die Vorschlüsselberechnungseinrichtung 34 einen 79-Bit-

25 Seitenvorschlüssel erzeugt. Hiervon geht das Ausführungsbeispiel von Fig. 7 für die Einrichtung zur Ermittlung des individuellen Schlüssels aus. Gemäß diesem Ausführungsbeispiel besteht die Einrichtung 36 aus einem Multiplexer 90 mit einem 79-Bit-Dateneingang, einem 4-Bit-Steuereingang und

30 einem 64-Bit-Datenausgang, wobei der Multiplexer 90 ausgebildet ist, um abhängig von der 4-Bit-Wortadresse an dem 4-Bit-Steuereingang 64 der 79 Bits des Seitenvorschlüssels an dem Dateneingang als 64-Bit-wortindividueller Schlüssel an den 64-Bit-Datenausgang auszugeben. Anders ausgedrückt, wird nach

35 dem Ausführungsbeispiel von Fig. 7 ein etwas längerer Seitenvorschlüssel generiert als das für die Datenverschlüsselung eigentlich erforderlich ist. In Abhängigkeit von der Wortad-

resse wird dann ein bestimmtes Segment aus diesem langen Seitenschlüssel ausgewählt und als wortindividueller Schlüssel genutzt.

- 5 Der Multiplexer 90 kann beispielsweise derart ausgebildet sein, dass er den Seitenvorschlüssel $\bar{K}_{Seite} = (k_0, k_1, k_2, \dots, k_{77}, k_{78})$ abhängig von der Wortadresse W auf den wortindividuellen 64-Bit-Schlüssel $\bar{K}_{Wort} = (k_0, k_1, \dots, k_{63})$ abbildet, wenn W=0000_b, auf $\bar{K}_{Wort} = (k_1, k_2, \dots, k_{64})$, wenn W=0001_b, auf \bar{K}_{Wort}
10 = $(k_2, k_3, \dots, k_{65})$, wenn W=0010_b ... und auf $\bar{K}_{Wort} = (k_{15}, k_{16}, \dots, k_{78})$, wenn W=1111_b.

Freilich sind auch andere Ausführungsbeispiele als die in Fig. 4 - 7 gezeigten für die Einrichtung 36 zur Ermittlung des wortindividuellen Schlüssels möglich. Die Einrichtung 36 könnte beispielsweise auch als eine kryptographische Einwegfunktion implementiert sein. Eine Einwegfunktion ist eine Funktion, bei der es wesentlich aufwendiger ist, die inverse Funktion zu ermitteln, oder bei der es unmöglich ist, dieselbe zu ermitteln. Ein Beispiel für eine Einwegfunktion ist beispielsweise eine modulare Operation, wie z.B. eine modulare Exponentiation. Diese Einwegfunktion lässt man dann auf die jeweilige Wortadresse wirken. Der so entstehende Funktionswert mit dem Seitenvorschlüssel verknüpft ergibt den wortindividuellen Schlüssel.
25

Nachdem sich die vorhergehenden Ausführungsbeispiele von Fig. 3 - 7 vornehmlich mit Ausführungsbeispielen für die Vorschlüsselberechnungseinrichtung 34 und die Ermittlungseinrichtung 36 beschäftigten, wird bezugnehmend auf Fig. 8 ein Ausführungsbeispiel für denjenigen Teil der Ver-/Entschlüsselungseinrichtung 16 beschrieben, der für die Entschlüsselung der von dem Speicher 12 (oder von einem Cache in dem Fall eines Treffers) zu der CPU 10 gesandten verschlüsselten Wörter auf der Basis des wortindividuellen Schlüssels, wie er von der Schlüsselerzeugungseinrichtung 18 zugesendet wird, zuständig ist.
30
35

Der Entschlüsselungsteil von Fig. 8 ist allgemein mit 16a gezeigt. Er umfasst einen Dateneingang 100 zum Empfang des verschlüsselten, zu entschlüsselnden Wortes von dem Speicher 5 12 sowie einen Datenausgang 102 zur Ausgabe des entschlüsselten Wortes zur Weiterleitung an die CPU 10. Ferner umfasst der Entschlüsselungsteil 16a einen Schlüsseleingang 104 zum Empfang des wortindividuellen Schlüssels von der Schlüsselerzeugungseinrichtung 18 (Fig. 1). Als interne Komponenten 10 umfasst der Empfangsteil 16a eine Permutationseinrichtung 106 zur Ausführung einer inversen Permutation, acht parallel geschaltete 4x4-S-Boxen S_1^{-1} - S_8^{-1} , die parallel geschaltet sind und jeweils unterschiedliche vier Bits eines 64-Bit-Wertes gemäß einer nichtlinearen Abbildung auf vier unterschiedliche 15 Bits eines Ausgangswertes abbilden, 108, eine 32-Bit-XOR-Verknüpfungseinrichtung 110, eine Rundenschlüsselgenerierungseinrichtung 112 und einen Schalter 114. Ein 32-Bit-Dateneingang der Permutationseinrichtung 106 ist mit dem Dateneingang 100 verbunden, um das verschlüsselte 32-Bit-Wort 20 zu erhalten. Die Permutationseinrichtung 106 permutiert die Bits des 32-Bit-Wortes am Permutationseingang und gibt das permutierte 32-Bit-Wort an seinem Permutationsausgang aus, wobei die Permutation zu einer vorbestimmten Permutation P invers ist, was mit P^{-1} angezeigt wird. Das Ergebnis der 25 Permutation gibt dieselbe als 32-Bit-Wert an die parallel geschalteten S-Boxen 108 aus. Jede S-Box S_1^{-1} - S_8^{-1} umfasst einen 4-Bit-Dateneingang und einen 4-Bit-Datenausgang. An jedem 4-Bit-Dateneingang der S-Boxen 108 liegen unterschiedliche vier Bits des 32-Bit-Wertes an, die von der Permutationseinrichtung 106 ausgegeben worden sind. Die S-Boxen 108 bilden, wie bereits erwähnt, durch eine nichtlineare Abbildung, die für jede der S-Boxen unterschiedlich sein kann und vorzugsweise auch ist, die vier Bitwerte an den Dateneingängen auf vier Bitwerte an den Datenausgängen ab. Die vier 30 Bitwerte an den Datenausgängen werden wieder zu einem 32-Bit-Wort zusammengefasst und einem Dateneingang der XOR-Verknüpfungseinrichtung 110 zugeführt. Die XOR- 35

Verknüpfungseinrichtung 110 umfasst einen weiteren Dateneingang. Dieser ist mit einem Ausgang der Rundenschlüsselgenerierungseinrichtung 112 verbunden, deren Eingang wiederum mit dem Schlüsseleingang 104 verbunden ist. Am Schlüsseleingang 5 104 liegt der wortindividuelle 64-Bit-Schlüssel an, wie er beispielsweise durch eine der Vorrichtungen von Fig. 4 - 7 erhalten worden ist oder durch eine andere mögliche Ausgestaltung der Schlüsselerzeugungseinrichtung 18. Die Rundenschlüsselgenerierungseinrichtung 112 erzeugt basierend auf 10 dem wortindividuellen Schlüssel einen 32-Bit-Rundenschlüssel. und gibt denselben an den zweiten Dateneingang der XOR-Verknüpfungseinrichtung 110 aus. Die XOR-Verknüpfungseinrichtung 110 verknüpft bitweise den 32-Bit-Rundenschlüssel und den 32-Bit-Ausgangswert von den S-Boxen 15 108, um an einem Datenausgang derselben ein 32-bit-Rundenzwischenergebnis zu erhalten. Ein Schaltereingang des Schalters 114 ist mit dem Datenausgang der XOR-Verknüpfungseinrichtung 110 verbunden. Der Schalter 114 umfasst zwei Schalterausgänge, nämlich einen 32-Bit- 20 Rundenfortführungsschalterausgang und einen 32-Bit-Rundenbeendigungsschalterausgang. Der Schalter 114 verbindet den Dateneingang mit dem Rundenfortführungsschalterausgang, so dass das verschlüsselte Wort im Dateneingang 100 ein vorbestimmte Anzahl an Malen die Einrichtungen 106 - 110 25 durchlaufen hat. Anzahl ist hoch genug, um eine ausreichende Sicherheit der Verschlüsselung sicherzustellen. Der Rundenfortführungsschalterausgang ist mit dem Permutationseingang der Permutationseinrichtung 106 verbunden, während der Rundenbeendigungsschalterausgang mit dem Datenausgang 102 verbunden ist, um nach der vorbestimmten Anzahl von Runden, die 30 das verschlüsselte Wort am Dateneingang 100 durchlaufen hat, das entschlüsselte 32-Bit-Wort am Datenausgang 102 auszugeben, der über den Datenbus 22 (Fig. 1) mit der CPU 10 verbunden ist. Die Rundenschlüsselgenerierungseinrichtung 112 35 ist derart ausgebildet, dass sie für jede Runde, die das verschlüsselte Wort die Einrichtungen 106 - 110 durchläuft,

einen unterschiedlichen Rundenschlüssel aus dem wortindividuellen 64-Bit-Schlüssel am Dateneingang 104 erzeugt.

Nachdem im Vorhergehenden der Aufbau des Entschlüsselungs-
5 teils 16a beschrieben worden ist, wird im Folgenden kurz
dessen Funktionsweise beschrieben. Das verschlüsselte 32-Bit-
Wort, das von dem Speicher 12 ausgelesen worden ist, erreicht
über den Dateneingang 100 die Permutationseinrichtung 106.
Diese permutiert das verschlüsselte Wort im Hinblick auf die
10 Anordnung bzw. Bitpositionsverteilung seiner Bits gemäß einer
Permutationsvorschrift P^{-1} . Daraufhin sorgen die parallel
geschalteten S-Boxen 108 für eine nichtlineare Abbildung des
permutierten 32-Bit-Wertes auf einen permuierten, abgebilde-
ten 32-Bit-Wert. Dieser wird mit einem ersten Rundenschlüs-
sel, den die Rundenschlüsselgenerierungseinrichtung 112 für
15 die erste Runde aus dem wortindividuellen Schlüssel erzeugt
hat, in der XOR-Verknüpfungseinrichtung 110 bitweise XOR-
verknüpft, wodurch das Rundenzwischenergebnis mit 32 Bits
erhalten wird. Dieses 32-Bit-Wort leitet der Schalter 114,
20 falls mehr als eine Runde durchgeführt werden soll, wieder an
den Eingang der Permutationseinrichtung 106 weiter, wodurch
die Permutation, die nichtlineare Abbildung sowie die XOR-
Verknüpfung wiederholt werden, letztere jedoch mit einem neu
ermittelten Rundenschlüssel. Nach der letzten Runde schaltet
25 der Schalter 114 auf den Rundenbeendigungsschalterausgang und
gibt das Rundenzwischenergebnis als entschlüsseltes 32-Bit-
Wort aus.

Der bezugnehmend auf Fig. 8 beschriebene Entschlüsselungsteil
30 der Ver-/Entschlüsselungseinrichtung 16 wird immer dann
aktiv, wenn die CPU verschlüsselte Speicherinhalte vom Spei-
cher 12 lädt. Fig. 9 zeigt ein Ausführungsbeispiel für einen
Verschlüsselungsteil 16b der Ver-/Entschlüsselungseinrichtung
16, der in der Lage ist, unverschlüsselte Wörter von der CPU
35 10 an den Speicher 12 bei einem Speichervorgang derart in
verschlüsselte Wörter zu verschlüsseln, dass sie, wenn sie
bei einem Ladevorgang wieder von der Ver-/Entschlüsselungs-

einrichtung 16 mittels des Entschlüsselungsteils 16a von Fig. 8 entschlüsselt werden, in ihrem ursprünglichen Zustand über den Datenbus 22 zur CPU 10 gelangen.

- 5 Der Verschlüsselungsteil 16b umfasst einen Dateneingang 120 zum Empfang eines unverschlüsselten, zu verschlüsselnden Wortes von der CPU 10 sowie einen Datenausgang 122 zur Ausgabe eines verschlüsselten Wortes zur Weiterleitung an den Speicher 12. Ferner umfasst der Verschlüsselungsteil 16b
10 einen Schlüsseleingang 124 zum Empfang des wortindividuellen Schlüssels. Darüber hinaus umfasst der Verschlüsselungsteil 16b eine Permutationseinrichtung 126 zur Permutierung eines 32-Bit-Wertes an einem Permutationseingang gemäß einer Permutation P, die zu der Permutation invers ist, die durch die
15 Permutationseinrichtung 106 durchgeführt wird, zu einem Permutationsergebnis an einem Permutationsausgang, acht parallel geschaltete 4x4-S-Boxen S₁ - S₈, 128, eine XOR-Verknüpfungseinrichtung 130, eine Rundengenerierungseinrichtung 132 und einen Schalter 134.
20 Die XOR-Verknüpfungseinrichtung 130 umfasst zwei 32-Bit-Dateneingänge, von denen der eine mit dem Dateneingang 120 der andere mit einem Datenausgang der Rundenschlüsselgenerierungseinrichtung 132 verbunden ist. Ein 32-Bit-Datenausgang der XOR-Verknüpfungseinrichtung 130 ist so mit den S-Boxen S₁ - S₈ verbunden, dass an den 4-Bit-Dateneingängen derselben jeweils vier unterschiedliche Bits des 32-Bit-Datenausgangs der XOR-Verknüpfungseinrichtung 130 anliegen. Die S-Boxen S₁ - S₈ bilden 4-Bit-Werte an ihren Dateneingängen gemäß nicht-linearen Abbildungen auf 4-Bit-Werte an ihren Datenausgängen ab, wobei die linearen Abbildungen zu denjenigen invers sind, die den S-Boxen von Fig. 8 bei 108 zugeordnet sind, d.h.
25 S_i(S_i⁻¹(x)) = S_i⁻¹(S_i(x)) = x für alle i = 1 ... 8 und für alle 4-Bit-Werte x.
30 Die 4-Bit-Werte an den Datenausgängen der S-Boxen 128 werden als 32-Bit-Wert an den Permutationseingang der Permutations-
35

einrichtung 126 weitergeleitet. Der Permutationsausgang der Permutationseinrichtung 126 ist mit einem Schaltereingang des Schalters 134 verbunden. Ein Rundenfortführungsschalterausgang des Schalters 134 ist mit dem ersten Dateneingang der 5 XOR-Verknüpfungseinrichtung 130 verbunden, während ein Rundenbeendigungsschalterausgang des Schalters 134 mit dem Datenausgang 122 verbunden ist. Ein Eingang der Rundenschlüsselgenerierungseinrichtung 132 ist mit dem Schlüsseleingang 124 verbunden.

10 Nachdem im Vorhergehenden der Aufbau des Verschlüsselungs- teils 16b beschrieben worden ist, wird im Folgenden dessen Funktionsweise beschrieben. Der Verschlüsselungsteil 16b ist im wesentlichen invers zu dem Entschlüsselungsteil 16a aufgebaut. Wenn ein unverschlüsseltes Wort am Dateneingang 120 zu 15 der XOR-Verknüpfungseinrichtung 130 gelangt, verknüpft die XOR-Verknüpfungseinrichtung 130 dieses unverschlüsselte Wort mit einem Rundenschlüssel, den die Rundenschlüsselgenerie- rungseinrichtung 132 aus dem wortindividuellen Schlüssel erzeugt. Dieser Rundenschlüssel ist derjenige Rundenschlüssel, den der Entschlüsselungsteil 16a bei seiner letzten 20 Runde verwenden wird, um das verschlüsselte Wort wieder zu entschlüsseln. Der sichergehende XOR-verknüpfte 32-Bit-Wert wird durch die S-Boxen 128 abgebildet auf einen abgebildeten 25 32-Bit-Wert. Diese Operation wird genau durch die S-Box- Abbildung der letzten Runde bei der Entschlüsselung in dem Entschlüsselungsteil 16a rückgängig gemacht werden. Der abgebildete 32-Bit-Wert wird von der Permutationseinrichtung 126 gemäß der Permutationsvorschrift P permutiert, um den 30 permuierten 32-Bit-Wert zu erhalten, der das Rundenzwischen- ergebnis darstellt. Diese Permutation der ersten Runde bei der Verschlüsselung wird bei der Entschlüsselung in der letzten Runde durch die Permutation P^{-1} in dem Entschlüsselungsteil 16a rückgängig gemacht werden. Solange weitere 35 Runden erwünscht sind, verbindet der Schalter 134 den Schaltereingang mit dem Rundenfortführungsschalterausgang, ansonsten mit dem Rundenbeendigungsschalterausgang, um das 32-Bit-

Rundenzwischenergebnis als das verschlüsselte Wort über den Datenausgang 122 an den Speicher 12 auszugeben. Die Runden-schlüssel, die die Rundenschlüsselgenerierungseinrichtung 132 aus dem wortindividuellen Schlüssel erzeugt, sind für die 5 jeweiligen Runden unterschiedlich und sind den Runden genau umgekehrt zugeordnet verglichen zu den Rundenschlüsseln, die die Rundenschlüsselgenerierungseinrichtung 112 für die Ent-schlüsselungsrunden erzeugt. Auf diese Weise ist sicherge-stellt, dass ein verschlüsseltes Wort, wie es durch den 10 Verschlüsselungsteil 16b erzeugt wird, durch den Entschlüs-selungsteil 16a wieder zu einem entschlüsselten Wort mit dem ursprünglichen Wert entschlüsselt wird. Der wortindividuelle Schlüssel, der an den Schlüsseleingängen 104 bzw. 124 an-liegt, ist bei Entschlüsselung und Verschlüsselung derselbe, 15 da ja sowohl beim Lade- als auch Speicherzugriff dieselbe eindeutige Adresse für das jeweilige Wort auf dem Adressbus 20 (Fig. 1) ausgegeben wird, so dass die Schlüsselerzeugungs-einrichtung 18 sowohl beim Laden als auch beim Speichern denselben wortindividuellen Schlüssel erzeugt.

20 Im Hinblick auf die Fig. 8 und 9 wird noch darauf hingewie-sen, dass es freilich auch viele andere Möglichkeiten der Realisierung der Ver-/Entschlüsselung in der Ver-/Entschlüs-selungseinrichtung 16 gibt. Unter anderem könnten Teile der 25 Komponenten in der Entschlüsselungseinheit 16a auch in der Verschlüsselungseinheit 16b verwendet werden, wenn entspre-chende Schalter je nach Ver- oder Entschlüsselung für eine geeignete Verschaltung dieser Komponenten sorgen. Ferner könnte Verschlüsselung und Entschlüsselung auch aus Doppel-runden bestehen, die eine Teilrunde mit einer S-Box-Abbildung 30 und eine Teilrunde mit einer inversen S^{-1} -Box-Abbildung aufweisen, so dass alle Komponenten der Ver-/Entschlüs-selungsvorrichtung sowohl beim Entschlüsseln als auch beim Verschlüsseln verwendet werden könnten.

35 Die vorhergehenden Ausführungsbeispiele gingen davon aus, dass als wortindividueller Schlüssel ein 64-BitSchlüssel der

Ver-/Entschlüsselungsvorrichtung zugeführt wird, die dann hierauf gemäß der Ausführungsbeispiele von Fig. 8 und 9 Rundenschlüssel erzeugt, die für die einzelnen Verschlüsselungs- bzw. Entschlüsselungsrunden verwendet werden. Das 5 nachfolgend bezugnehmend auf Fig. 10 beschriebene Ausführungsbeispiel unterscheidet sich von diesen Ausführungsbeispielen dadurch, dass als wortindividueller Schlüssel eine Rundenschlüsselfolge der Ver-/Entschlüsselungseinrichtung zugeführt wird, die aus Rundenschlüsseln besteht, die für die 10 Wörter einer Seite zwar gleich sind aber in unterschiedlicher Reihenfolge angeordnet sind.

Fig. 10 zeigt ein Ausführungsbeispiel für eine einen derartigen wortindividuellen Schlüssel erzeugende Einrichtung 36''.
15 Die Einrichtung 36'' umfasst eine Rundenschlüsselberechnungseinrichtung 140, eine Nachschlagtabelle 142, eine Ausleseeinrichtung 144 und eine Auswahlreihenfolgebestimmungseinrichtung 146. Die Rundenschlüsselberechnungseinrichtung 140 umfasst einen Dateneingang und einen Datenausgang, wobei am
20 Dateneingang der 64-Bit-Seitenvorschlüssel vorliegt, wie er entweder von der Vorschlüsselberechnungseinrichtung 34 oder dem Zwischenspeicher 38 zugeführt wird (siehe Fig. 2a). Die Rundenschlüsselberechnungseinrichtung 140 berechnet auf der Basis des Seitenvorschlüssels mehrere unterschiedliche Rundenschlüssel. Die Anzahl der Rundenschlüssel richtet sich
25 nach der Anzahl der Runden der iterativen Blockchiffre, die durch die Ver-/Entschlüsselungseinrichtung 16 implementiert wird, wie z.B. durch den Entschlüsselungs- und Verschlüsselungsteil der Fig. 8 und 9. Für jede Runde trägt also die
30 Rundenschlüsselberechnungseinrichtung 140 einen unterschiedlichen Rundenschlüssel in die Nachschlagtabelle 142 ein. Die Auswahlreihenfolgebestimmungseinrichtung 146 empfängt an einem Dateneingang die 4-Bit-Wortadresse der gerade auf dem Adressbus 20 befindlichen eindeutigen Adresse 30. Je nach
35 Wortadresse wählt die Auswahlreihenfolgebestimmungseinrichtung 146 eine vorbestimmte unterschiedliche Reihenfolge bezüglich der Rundenschlüssel aus, mit der dieselben ausgele-

sen werden sollen. Diese Reihenfolge zeigt dieselbe der Ausleseeinrichtung 144 an, die darauf hin gemäß der angezeigten Reihenfolge die 32-Bit-Langen-Rundenschlüssel ausliest, um dieselben in der angezeigten Reihenfolge als Rundenschlüssel 5 selfolge von 32-Bit-Rundenschlüsseln an die Ver-/Entschlüsselungseinrichtung 16 auszugeben. Unterschiedliche Wörter mit unterschiedlichen Wortadressen in der selben Seite führen folglich auch bei selbem Wortinhalt zu einem unterschiedlichen Chifferrat, da bei der iterativen Blockchiffre in der Ver-/Entschlüsselungseinrichtung 16 die Rundenschlüssel 10 folge bei den einzelnen Runden nicht dieselbe ist.

Anders ausgedrückt, wird gemäß dem Ausführungsbeispiel von Fig. 10 von einer iterativen Blockchiffre bei der Ver-/Entschlüsselungseinrichtung 16 ausgegangen, bei welcher in 15 jeder Runde ein eigener Rundenschlüssel angewandt wird. Diese Rundenschlüssel stehen in Registern der Nachschlagetabelle 142 bereit, nachdem sie vorab aus dem Seitenvorschlüssel berechnet worden sind. Die 4-Bit-Wortadresse bestimmt nun die 20 Reihenfolge, in welcher die Rundenschlüssel bei der iterativen Blockchiffre angewendet werden.

Die im Vorhergehenden beschriebenen Ausführungsbeispiele zur Generierung von Schlüsseln für die Verschlüsselung zu speichernder Daten und/oder Entschlüsselung von gespeicherter, 25 gelesener Daten beim Zugriff auf einen Speicher liefern folglich eine adressabhängige Bereichsschlüsselgenerierung. Anstatt jedoch die Bereichsschlüsselgenerierung für jedes Wort gleichermaßen kompliziert durchzuführen, wird der Prozess der Bereichsschlüsselgenerierung in zwei Teilprozesse 30 aufgespalten, nämlich einen relativ aufwendigen und langsamen Schritt und einen einfachen und schnellen Schritt, der praktisch gratis ist. Nur der einfache Schritt muss für jedes einzelne Wort ausgeführt werden, der aufwendige Schritt 35 dagegen nur einmal für mehrere Worte gleichzeitig.

Dabei ist das Festhalten an der adressabhängigen Bereichsschlüsselgenerierung mehr als nur sinnvoll: In einem Mikroprozessor beträgt die Wortgröße heutzutage einige wenige Byte. Zum Beispiel 4 Byte bzw. 32 Bit. Eine kryptographische 5 32-Bit-Blockchiffre macht aber keinen Sinn. Die Zahl 32 ist klein genug, so dass ein Unbefugter für alle möglichen $2^{32} \approx 4,3$ Milliarden Klartexte die zugehörigen Chiffretexte sammeln und in einer Art Codier-Lexikon auflisten könnte. Kryptographische Blockchiffre machen erst für Blockbreiten von 64 10 Bit, besser für 128 Bit, Sinn. Dieses Problem kann auch nicht durch Anwendung eines CBC-Modus, wie er in der Beschreibungseinleitung beschrieben worden ist, auf eine „32-Bit-Blockchiffre“ gelöst werden. Die adressabhängige Bereichsschlüsselgenerierung löst dieses Problem dagegen befriedigend. Nun kann kein Codier-Lexikon obiger Art mehr erstellt 15 werden. Denn das selbe 32-Bit-Klartext-Wort, das an zwei unterschiedlichen Speicheradressen vorkommt, wird mit verschiedenen Bereichsschlüsseln verschlüsselt. Damit werden die zugehörigen Chiffretexte ebenfalls verschieden sein, auch 20 wenn der zugrundeliegende Klartext derselbe ist.

Die Bereichsschlüsselgenerierung zum Zugriff auf einen Speicher mit verschlüsseltem Inhalt gemäß obiger Ausführungsbeispiele bestand darin, aus dem geheimen Generalschlüssel und 25 der Seitenadresse bzw. Pageadresse einen pageweit bzw. seitenweit gültigen Vorschlüssel zu generieren. Das ist der aufwendige Schritt, der nur einmal pro Page bzw. Seite durchgeführt werden muss. Dann wird aus dem Vorschlüssel und der Wortadresse auf einfache Weise ein wortindividueller Schlüssel 30 hergeleitet. Die Verschlüsselung des Wortes erfolgt nun mit dem wortindividuellen Schlüssel.

Die sich ergebenden Vorteile sind die Folgenden: Die Berechnung des Pageschlüssels muss bestimmte kryptographische 35 Kriterien erfüllen und ist entsprechend aufwendig. Der Pageschlüssel wird entweder in einer eigenen Hardwareeinheit berechnet oder es wird die Verschlüsselungseinheit 16 auch

- zur Berechnung des Pageschlüssels genutzt. Da nun die Berechnung des Pageschlüssels seltener erforderlich ist (nur einmal pro Page) kann aus diesem Grund die Hardwareeinheit für die Pageschlüsselgenerierung kleiner gestaltet werden. Im anderen
- 5 Fall, bei dem die Verschlüsselungshardware zur Pageschlüsselgenerierung mitbenutzt wird, steigt in Folge der seltener gewordenen Beanspruchung der Verschlüsselungshardware die Verschlüsselungsrate.
- 10 Bezugnehmend auf obige Ausführungsbeispiele wird noch darauf hingewiesen, dass beispielsweise ohne weiteres die XOR-Gatter durch NXOR-Gatter ersetzt werden könnten. Die im vorhergehenden beschriebene vorübergehende Speicherung der Seitenvorschlüssel mit Verdrängungsstrategie kann auch durch eine
- 15 Speicherung in einem ausreichend großen flüchtigen Speicher ohne Verdrängungsstrategie ersetzt werden, so dass die Seitenvorschlüssel bei fehlender Energiezufuhr automatisch gelöscht würden. Ferner könnte vor der Generierung des Seitenvorschlüssels die Seitenadresse auch anderen Operationen
- 20 unterzogen werden als die Expansion von Fig. 3. Ferner muss die vorliegende Erfindung nicht basierend auf den kleinsten adressierbaren Einheiten des direkt adressierbaren Speichers arbeiten, sondern es können auch größere adressierbare Einheiten gewählt werden. Ferner könnte die Abbildung der Seitenadressen auf die Seitenvorschlüssel jegliche Abbildung sein, vorzugsweise natürlich eine nichtlineare Abbildung.
- 25

Insbesondere wird darauf hingewiesen, dass abhängig von den Gegebenheiten das erfindungsgemäße Schema auch in Software 30 implementiert sein kann. Die Implementation kann auf einem digitalen Speichermedium, insbesondere einer Diskette oder einer CD mit elektronisch auslesbaren Steuersignalen erfolgen, die so mit einem programmierbaren Computersystem zusammenwirken können, dass das entsprechende Verfahren ausgeführt 35 wird. Allgemein besteht die Erfindung somit auch in einem Computerprogrammprodukt mit auf einem maschinenlesbaren Träger gespeicherten Programmcode zur Durchführung des erfin-

dungsgemäßen Verfahrens, wenn das Computerprogrammprodukt auf einem Rechner abläuft. In anderen Worten ausgedrückt kann die Erfindung somit als ein Computerprogramm mit einem Programmcode zur Durchführung des Verfahrens realisiert werden, wenn
5 das Computerprogramm auf einem Computer abläuft.

Bezugszeichenliste

- 10 CPU
12 Speicher
5 12a Seite
12b Wort
14 Zugriffssystem
16 Ver-/Entschlüsselungseinrichtung
16a Entschlüsselungsteil
10 16b Verschlüsselungsteil
18 Schlüsselerzeugungseinrichtung
20 Adressbus
22 Datenbus
30 eindeutige Adresse
15 30a Seitenadressteil
30b Wortadressteil
32 Seitenwechselfeststellungseinrichtung
34 Vorschlüsselberechnungseinrichtung
36 Einrichtung zur Ermittlung des individuellen
20 Schlüssels
38 Zwischenspeicher
40 Verweis
50 Dateneingang
52 Datenausgang
25 54 Expansionseinrichtung
56 Blockchiffremodul
58 expandierter Datenblock
60a-60e XOR-Verknüpfungseinrichtungen
70 Nachschlagtabelle
30 72 XOR-Verknüpfungseinrichtung
74 Index
80 Nachschlagtabelle
82 steuerbare Permutationseinrichtung
84 Index
35 90 Multiplexer
100 Dateneingang
102 Datenausgang

104 Schlüsseleingang
106 Permutationseinrichtung
108 S-Boxen
110 XOR-Verknüpfungseinrichtung
5 112 Rundenschlüsselgenerierungseinrichtung
114 Schalter
120 Dateneingang
122 Datenausgang
124 Schlüsseleingang
10 126 Permutationseinrichtung
128 S-Boxen
130 XOR-Verknüpfungseinrichtung
132 Rundenschlüsselgenerierungseinrichtung
134 Schalter
15 140 Rundenschlüsselberechnungseinrichtung
142 Nachschlagtabelle
144 Ausleseeinrichtung
146 Auswahlreihenfolgebestimmungseinrichtung

Patentansprüche

1. Vorrichtung zur Erzeugung eines individuellen Schlüssels zum Zugriff auf eine vorbestimmte adressierbare Einheit (12b) eines in adressierbaren Einheiten (12b) gegliederten Speichers (12), wobei die adressierbaren Einheiten (12b) gruppenweise zu Seiten (12a) zusammengefasst sind, wobei der vorbestimmten adressierbaren Einheit eine eindeutige Adresse (30) zugeordnet ist, die sich aus einer Seitenadresse (30a), die die Seite angibt, der die adressierbare Einheit angehört, und einer Einheitsadresse (30b), die die adressierbare Einheit unter den übrigen adressierbaren Einheiten, die der Seite angehören, identifiziert, zusammensetzt, mit folgenden Merkmalen:
15 eine Einrichtung (34) zum Berechnen eines Seitenvorschlüssels auf der Basis der Seitenadresse (30a), und einer Einrichtung (36) zum Ermitteln des individuellen Schlüssels auf der Basis des Seitenvorschlüssels und der Wortadresse (30b).
2. Vorrichtung gemäß Anspruch 1, bei der die Einrichtung (34) zum Berechnen des Seitenvorschlüssels mehr Chipfläche und/oder mehr Verarbeitungslaufzeitdauer aufweist als die Einrichtung (36) zum Ermitteln des individuellen Schlüssels.
3. Vorrichtung gemäß Anspruch 1 oder 2, bei der die Einrichtung (34) zum Berechnen eine Einrichtung (56) zum Chiffrieren der Seitenadresse oder eines aus derselben hergeleiteten Datenblocks mit einem Generalschlüssel aufweist, um den Seitenvorschlüssel zu halten.
4. Vorrichtung gemäß Anspruch 3, bei der die Einrichtung (56) zum Chiffrieren eine Einrichtung (54) zum Expandieren der Seitenadresse (30a) aufweist, um einen expandierten Datenblock (58) zu erhalten, durch Abbilden jedes Bits der

Seitenadresse (30a) auf zumindest ein unterschiedliches Bit des expandierten Datenblocks (58) und Abbilden zumindest eines Bits der Seitenadresse (30a) auf zumindest zwei Bits des expandierten Datenblocks (58), und bei der die Einrichtung zum Chiffrieren angepasst ist, um den expandierten Datenblock (58) zu chiffrieren.

5. Vorrichtung gemäß einem der Ansprüche 1 bis 4, bei der die Einrichtung (36) zum Ermitteln des individuellen Schlüssels eine Einrichtung (60a - 60e) zur bitweisen XOR-Verknüpfung von Bits der Wortadresse (30b) mit vorbestimmten Bits des Seitenvorschlüssels aufweist, um den individuellen Schlüssel zu erhalten.
- 10 15. 6. Vorrichtung gemäß einem der Ansprüche 1 bis 4, bei der die Einrichtung (36) zum Ermitteln des individuellen Schlüssels ferner folgende Merkmale aufweist:
 - 20 eine Nachschlagtabelle (70) mit Maskierungsvektoren, von denen jeder einer möglichen Wortadresse zugeordnet ist;
 - 25 einer Einrichtung (74) zum Nachschlagen in der Nachschlagtabelle mit der Wortadresse als Index, um den zugeordneten Maskierungsvektor zu erhalten; und
 - 30 35. einer Einrichtung (72) zur bitweisen XOR-Verknüpfung des Maskierungsvektors und des Seitenvorschlüssels, um den individuellen Schlüssel zu erhalten.
 - 30 7. Vorrichtung gemäß einem der Ansprüche 1 bis 4, bei der die Einrichtung zum Ermitteln des individuellen Schlüssels eine steuerbare Permutationseinrichtung (82) zum Permutieren von Bits der Seitenadresse gemäß einer Permutationsvorschrift, die von der Wortadresse abhängt, um den individuellen Schlüssel zu erhalten, aufweist.

8. Vorrichtung gemäß einem der Ansprüche 1 bis 4, bei der die Einrichtung (36) zum Ermitteln des individuellen Schlüssels eine Auswahleinrichtung (90) zum Auswählen von Bits der Seitenadresse gemäß einer Auswahlvorschrift, die von der Wortadresse abhängt, um den individuellen Schlüssel zu erhalten, aufweist.
9. Vorrichtung gemäß einem der Ansprüche 1 bis 4, bei der die Einrichtung (36) zum Ermitteln des individuellen Schlüssels folgendes Merkmal aufweist:
- eine Einrichtung (140) zum Herleiten mehrerer Rundenschlüssel aus dem Seitenvorschlüssel; und
- 15 10. eine Einrichtung (144, 146) zum Definieren einer Reihenfolge unter den Rundenschlüsseln, um eine Folge der Rundenschlüssel zu erhalten, wobei die Reihenfolge von der Wortadresse abhängt, und die Folge den individuellen Schlüssel darstellt.
- 20 10. Vorrichtung gemäß einem der vorhergehenden Ansprüche, die ferner folgende Merkmale aufweist:
- eine Einrichtung (38) zum Speichern des berechneten Seitenvorschlüssels; und
- 25 eine Einrichtung zum Überprüfen, ob bei einem nächsten Zugriff auf eine weitere vorbestimmte Einheit, der eine weitere eindeutige Adresse zugeordnet ist, in dem temporären Speicher (38) ein bereits berechneter Seitenvorschlüssel
- 30 existiert, der basierend auf einer Seitenadresse einer eindeutigen Adresse berechnet worden ist, die zu der Seitenadresse der weiteren eindeutigen Adresse identisch ist, und, falls dies der Fall ist, zum Weiterleiten, unter Umgehung der Einrichtung (34) zum Berechnen, des bereits berechneten
- 35 Seitenvorschlüssels an die Einrichtung (36) zum Ermitteln und, falls dies nicht der Fall ist, zum Weiterleiten der

Seitenadresse der weiteren eindeutigen Adresse an die Einrichtung (34) zum Berechnen.

11. System zum Zugriff auf eine vorbestimmte adressierbare
5 Einheit eines in adressierbaren Einheiten gegliederten Speichers (12), wobei die adressierbaren Einheiten (12b) gruppenweise zu Seiten (12a) zusammengefasst sind, wobei der vorbestimmten adressierbaren Einheit eine eindeutige Adresse (30) zugeordnet ist, die sich aus einer Seitenadresse (30a), die
10 die Seite angibt, der die adressierbare Einheit angehört, und einer Einheitsadresse (30b), die die adressierbare Einheit unter den übrigen adressierbaren Einheiten, die der Seite angehören, identifiziert, zusammensetzt, mit folgenden Merkmalen:

15 einer Vorrichtung zur Erzeugung gemäß einem der Ansprüche 1 bis 10; und

20 einer Vorrichtung zum Entschlüsseln eines verschlüsselten Speicherinhalts in der vorbestimmten adressierbaren Einheit auf der Basis des individuellen Schlüssels.

12. System zum Zugriff auf eine vorbestimmte adressierbare Einheit eines in adressierbaren Einheiten gegliederten Speichers (12), wobei die adressierbaren Einheiten (12b) gruppenweise zu Seiten (12a) zusammengefasst sind, wobei der vorbestimmten adressierbaren Einheit eine eindeutige Adresse (30) zugeordnet ist, die sich aus einer Seitenadresse (30a), die die Seite angibt, der die adressierbare Einheit angehört, und
25 einer Einheitsadresse (30b), die die adressierbare Einheit unter den übrigen adressierbaren Einheiten, die der Seite angehören, identifiziert, zusammensetzt, mit folgenden Merkmalen:

35 einer Vorrichtung zur Erzeugung gemäß einem der Ansprüche 1 bis 10; und

eine Vorrichtung zum Verschlüsseln zu speichernder Daten auf der Basis des individuellen Schlüssels und Schreiben der verschlüsselten, zu schreibenden Daten in die vorbestimmte adressierbare Einheit.

5

13. Verfahren zur Erzeugung eines individuellen Schlüssels zum Zugriff auf eine vorbestimmte adressierbare Einheit (12b) eines in adressierbaren Einheiten (12b) gegliederten Speichers (12), wobei die adressierbaren Einheiten (12b) gruppenweise zu Seiten (12a) zusammengefasst sind, wobei der vorbestimmten adressierbaren Einheit eine eindeutige Adresse (30) zugeordnet ist, die sich aus einer Seitenadresse (30a), die die Seite angibt, der die adressierbare Einheit angehört, und einer Einheitsadresse (30b), die die adressierbare Einheit unter den übrigen adressierbaren Einheiten, die der Seite angehören, identifiziert, zusammensetzt, mit folgenden Schritten:

Berechnen eines Seitenvorschlüssels auf der Basis der Seiten-
20 Adresse (30a), und

Ermitteln des individuellen Schlüssels auf der Basis des Seitenvorschlüssels und der Wortadresse (30b).

25 14. Computer-Programm mit einem Programmcode zur Durchführung des Verfahrens nach Anspruch 13, wenn das Computer-Programm auf einem Computer abläuft.

30

1/9

FIG 1

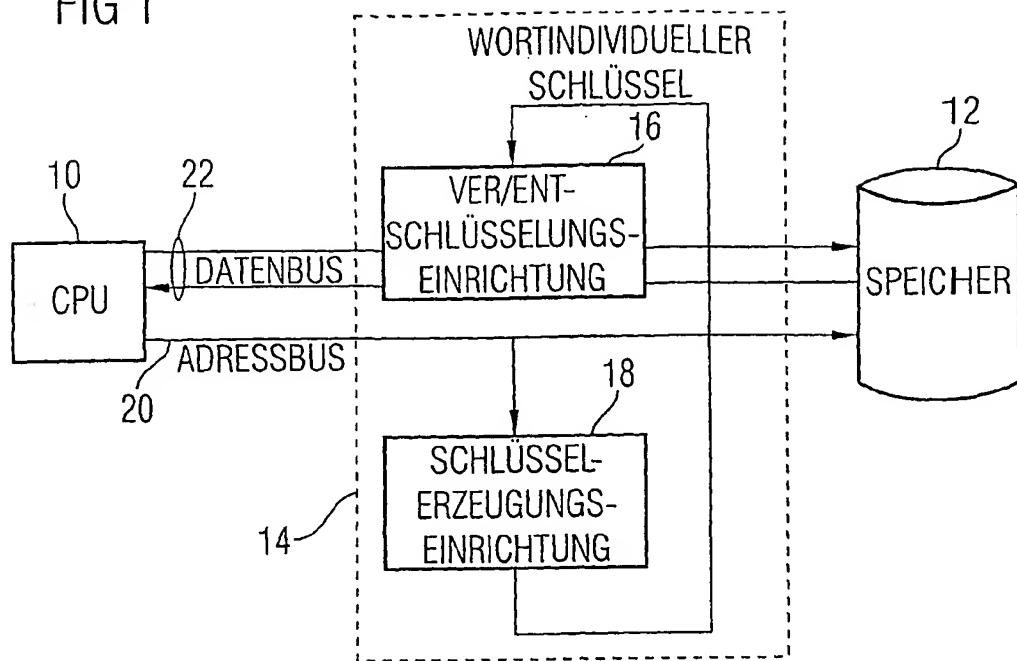


FIG 2B

WORTADRESSE/ VERSATZWERT

WORT- NUMMER

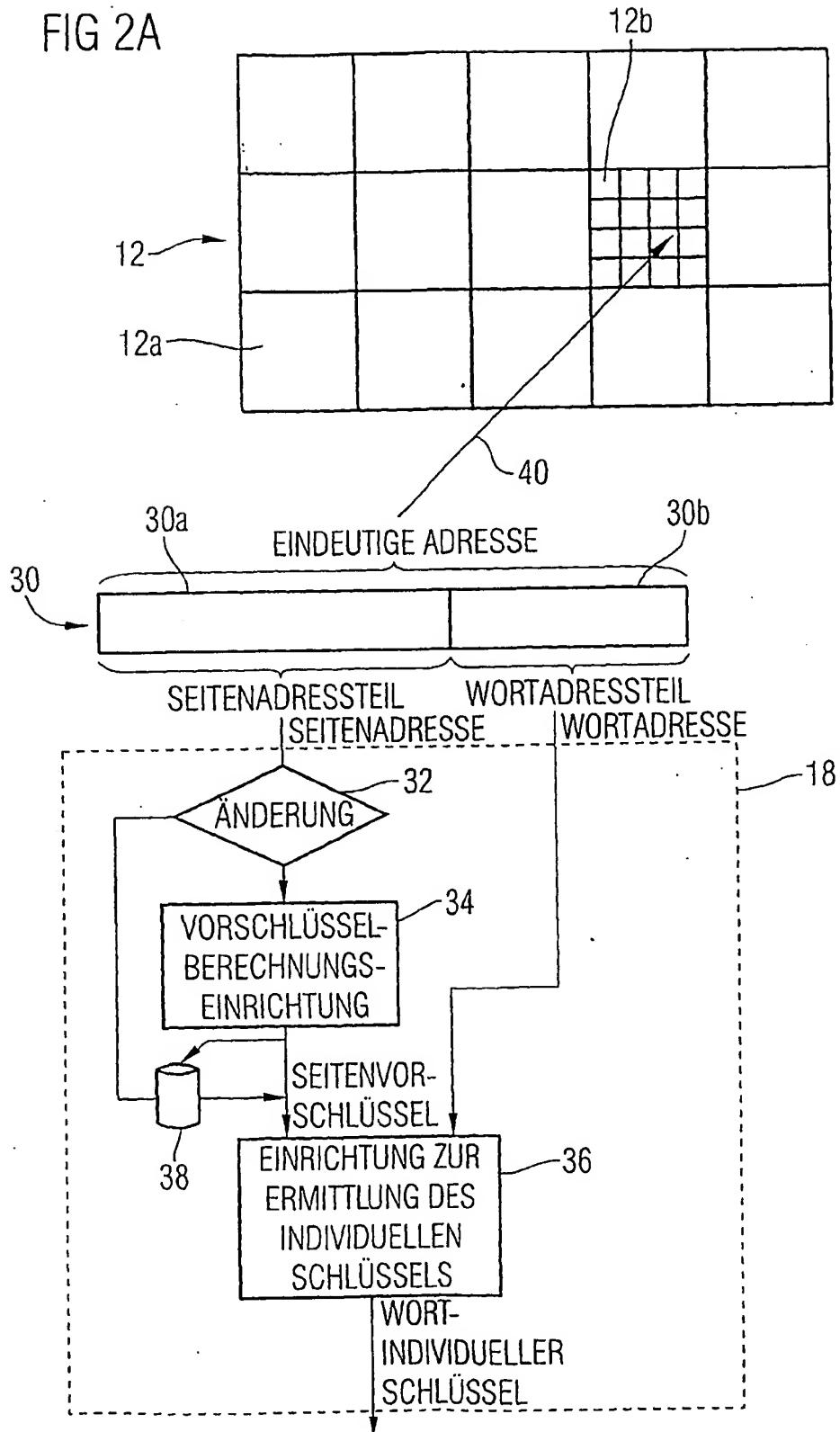
12b

12a

0000 WORT_0	0001 WORT_1	0010 WORT_2	0011 WORT_3
0100 WORT_4	0101 WORT_5	0110 WORT_6	0111 WORT_7
1000 WORT_8	1001 WORT_9	1010 WORT_10	1011 WORT_11
1100 WORT_12	1101 WORT_13	1110 WORT_14	1111 WORT_15

2/9

FIG 2A



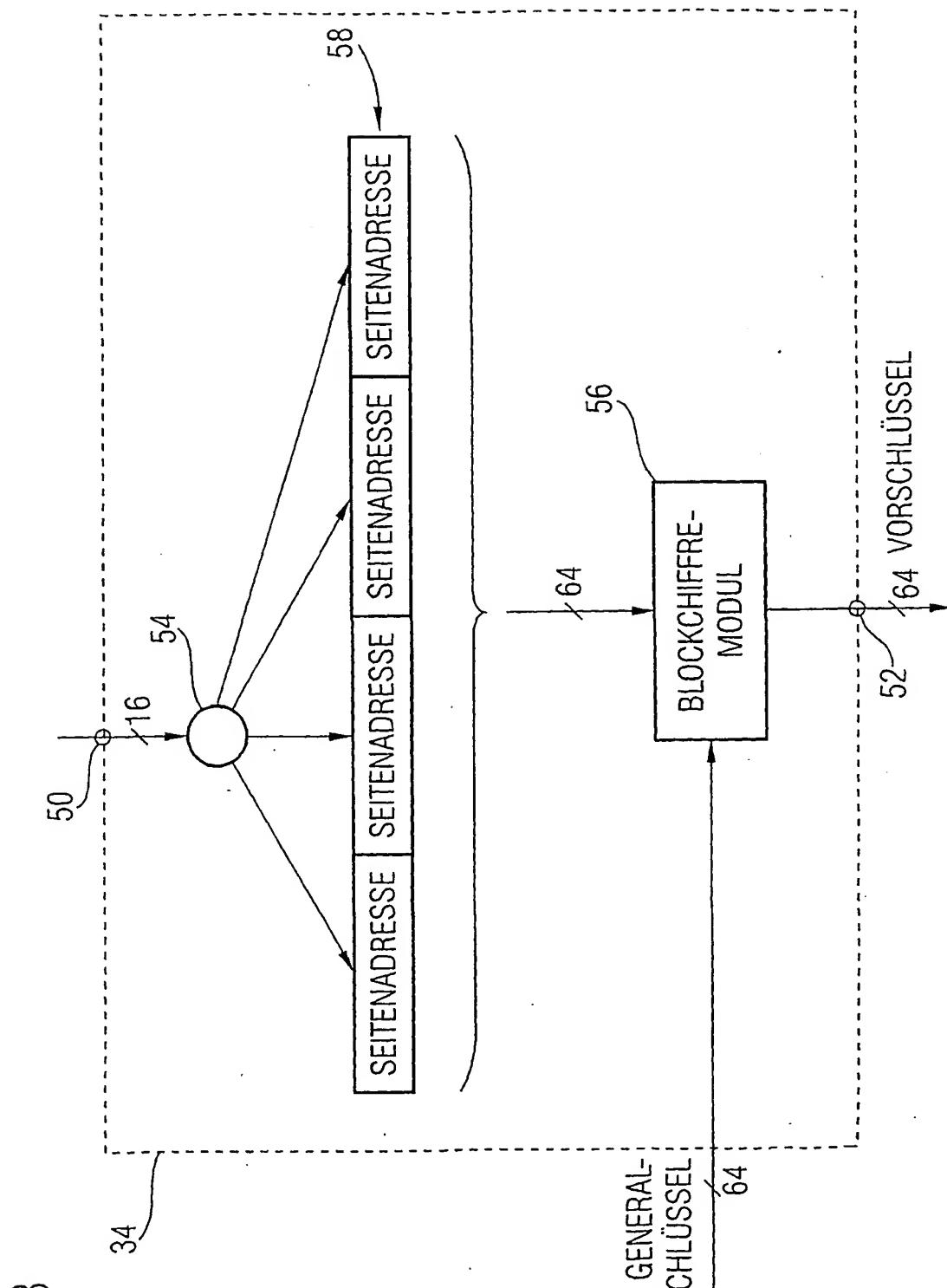
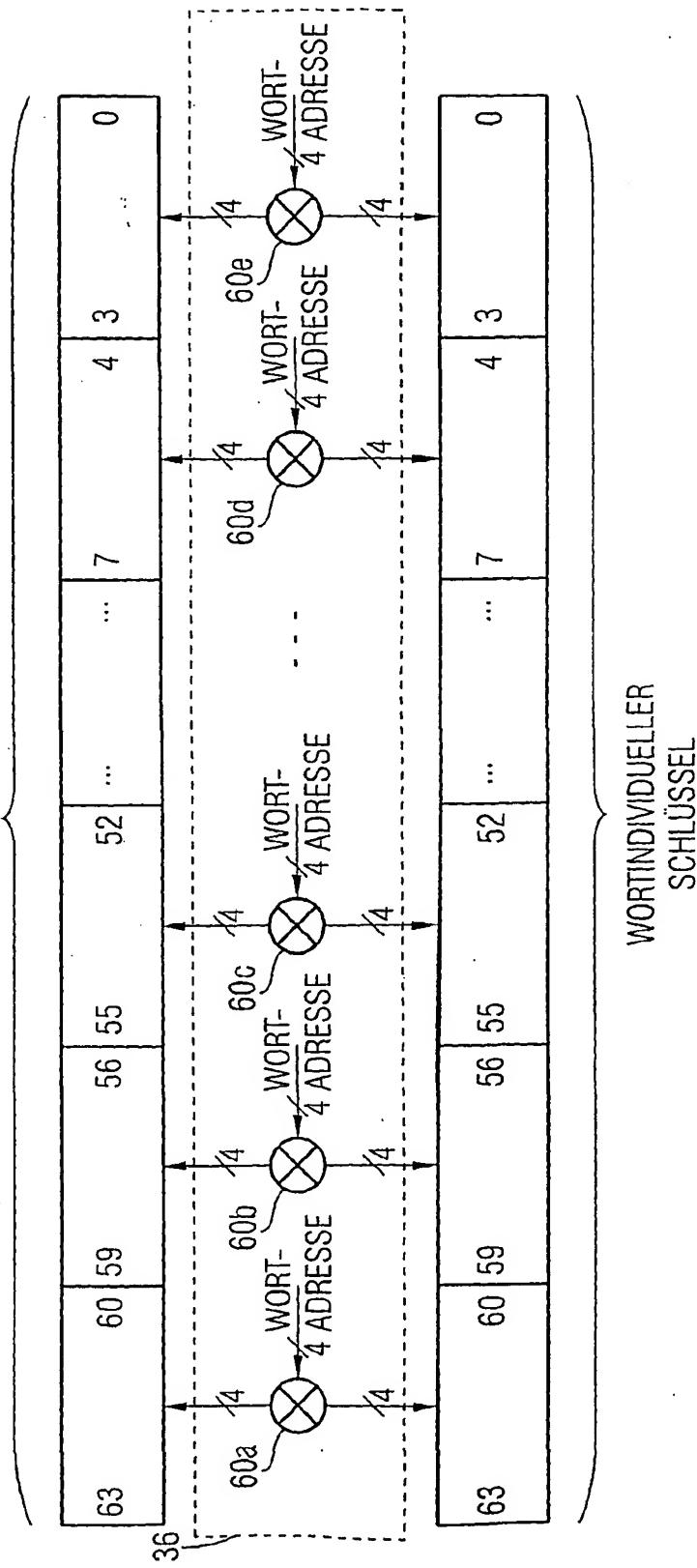


FIG 3

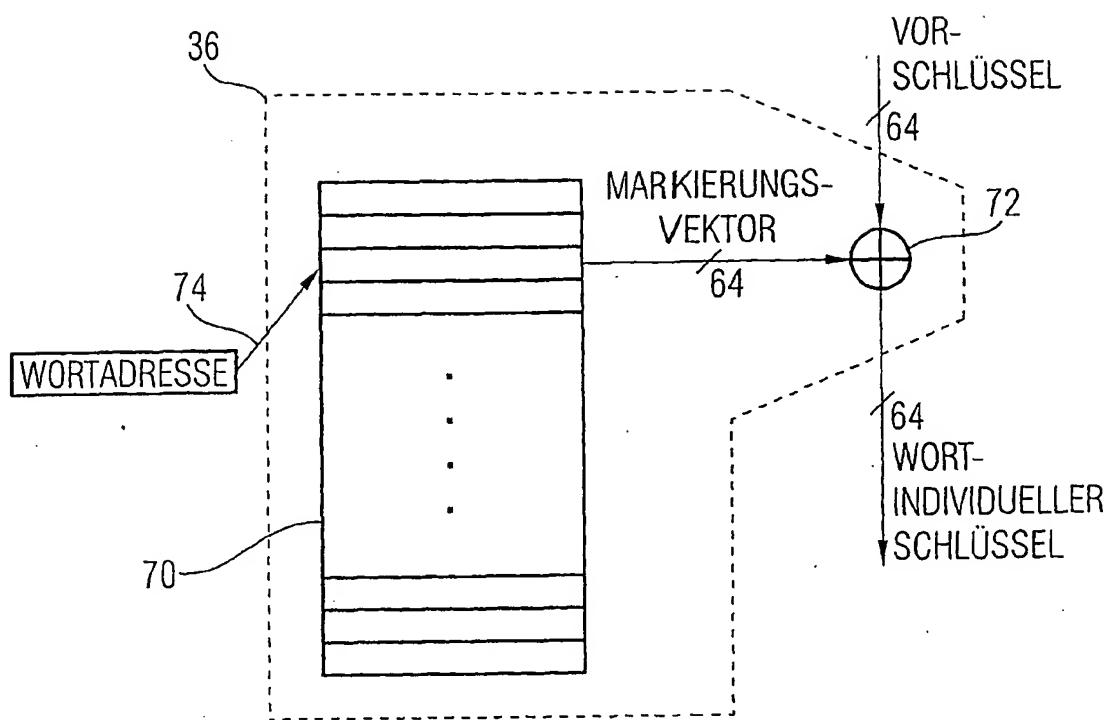
4/9

FIG 4



5/9

FIG 5



6/9

FIG 6

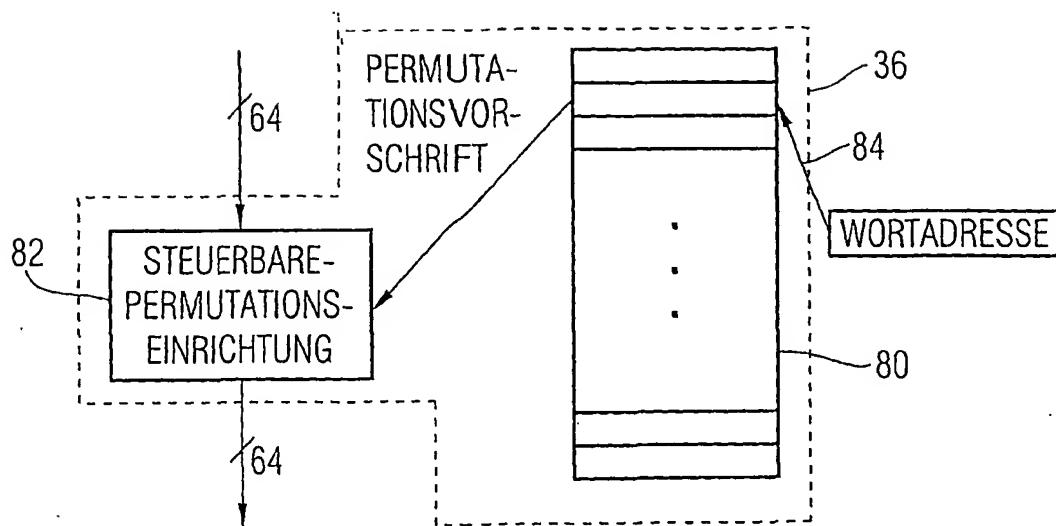
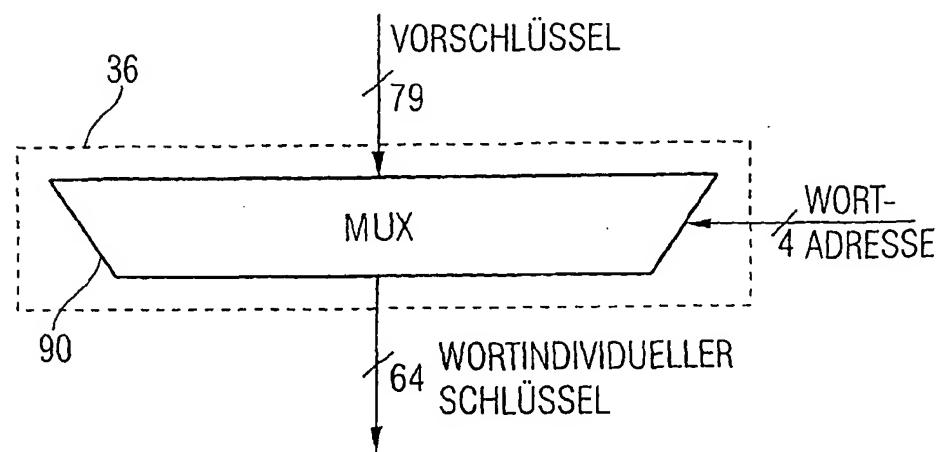
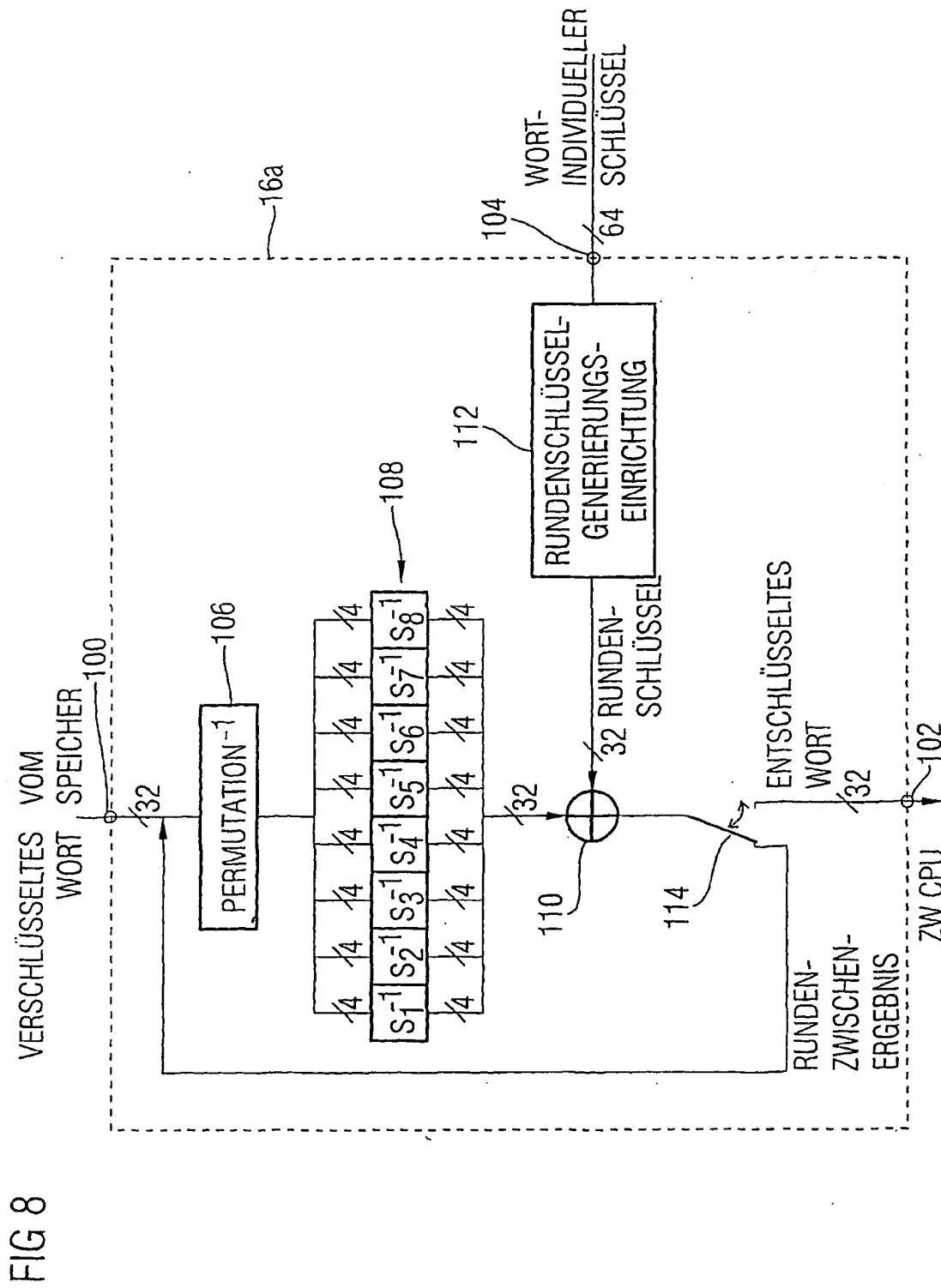


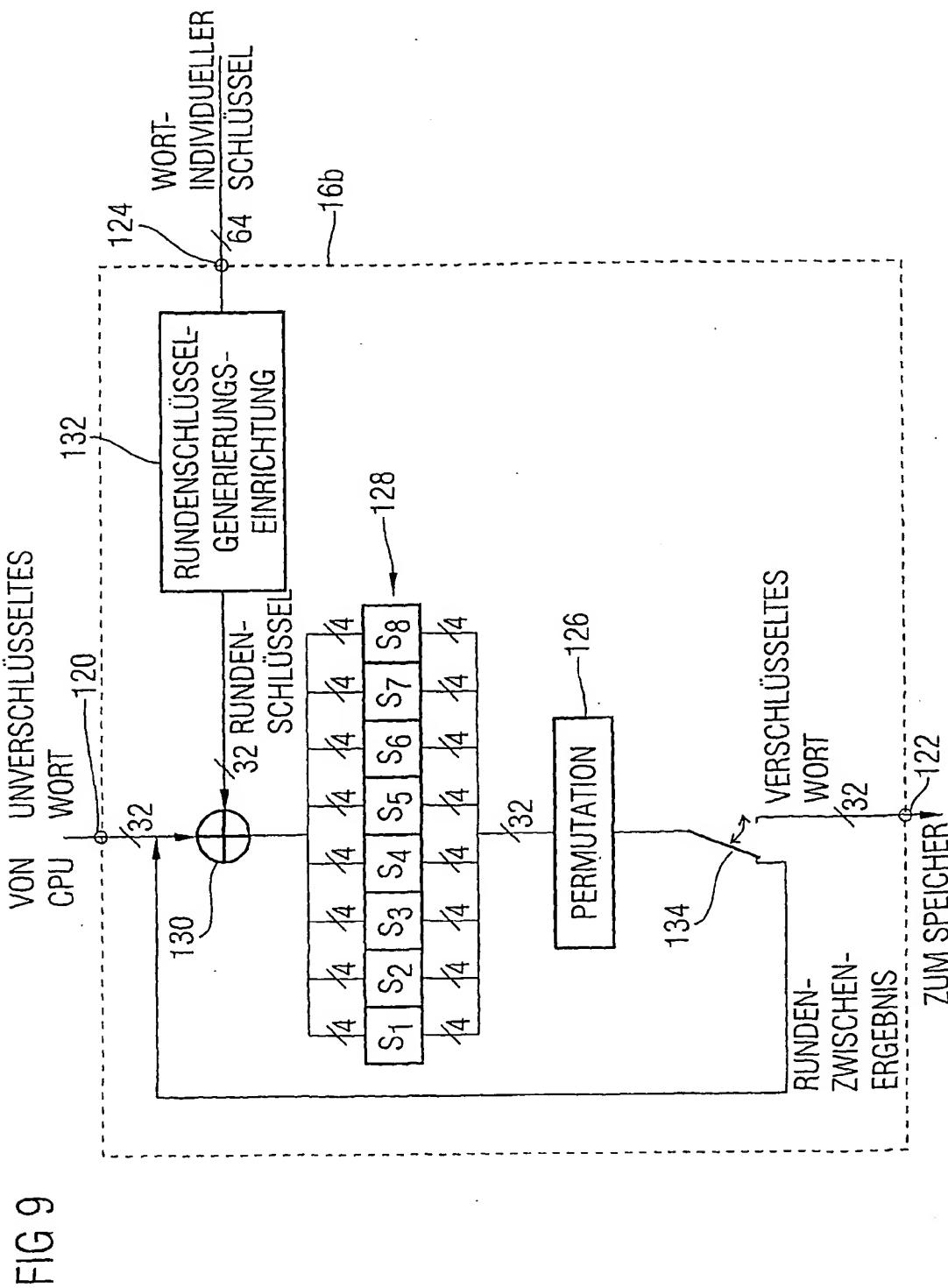
FIG 7



7/9



8/9



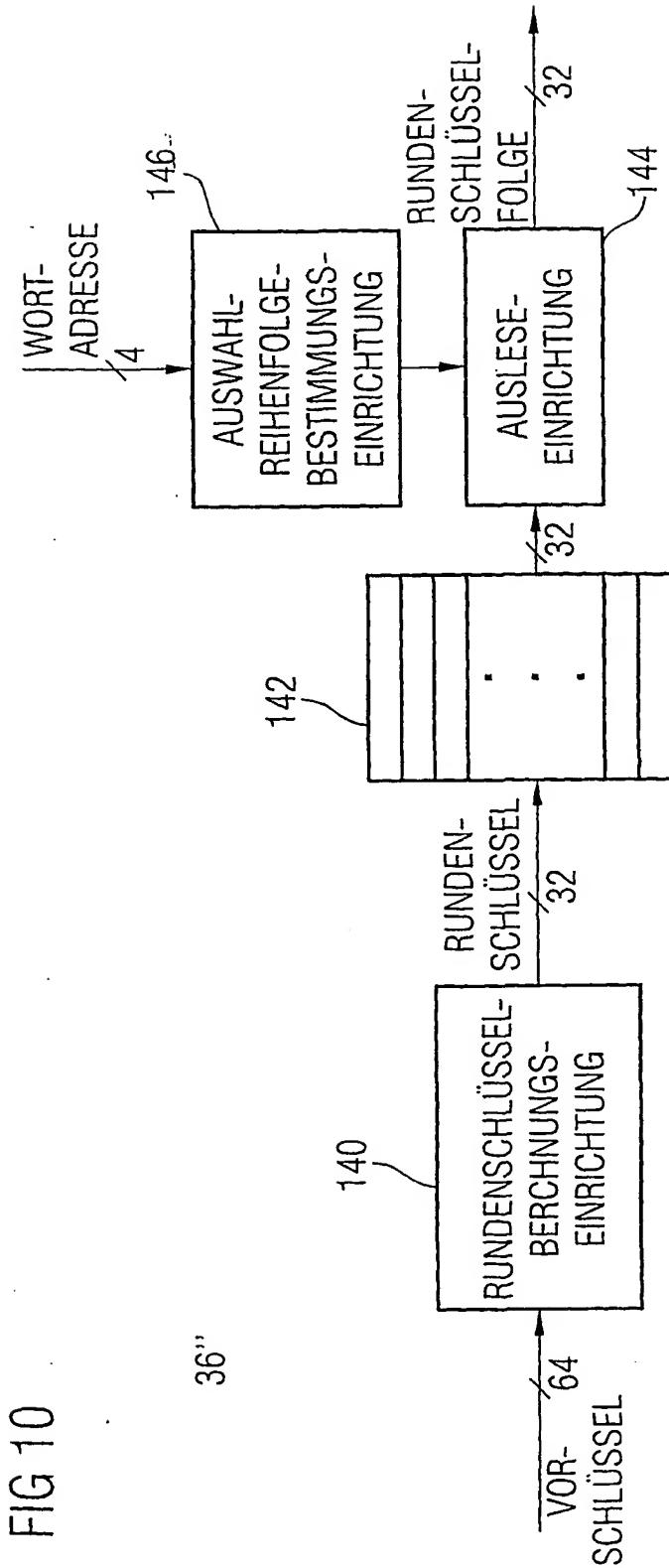


FIG 10